Applied Research Laboratory

Technical Report

STABILITY EXPLOITATION AND SUBSPACE ARRAY PROCESSING

by

M. E. Kotanchek

PENNSTATE

1 8 5 5

The Pennsylvania State University
APPLIED RESEARCH LABORATORY
P.O. Box 30
State College, PA 16804

STABILITY EXPLOITATION AND SUBSPACE ARRAY PROCESSING

by

M. E. Kotanchek

Technical Report No. TR 96-001
January 1996

Supported by:
Office of Naval Research

L.R. Hettche, Director
Applied Research Laboratory

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. AGENCY USE ONLY *(Leave blank)*

2. REPORT DATE
January 1996

3. REPORT TYPE AND DATES COVERED

4. TITLE AND SUBTITLE
Stability Exploitation and Subspace Array Processing

5. FUNDING NUMBERS
N00039-92-C-0100

6. AUTHOR(S)
M. E. Kotanchek

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Applied Research Laboraatory
The Pennsylvania State University
P. O. Box 30
State College, PA 16804

8. PERFORMING ORGANIZATION REPORT NUMBER
TR-96-001

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
Office of Naval Research
Ballstown Tower 1
800 North Quncy Street
Arlington, VA 22217-5660

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES

12a. DISTRIBUTION/AVAILABILITY STATEMENT
Unlimited

12b. DISTRIBUTION CODE

13. ABSTRACT *(Maximum 200 words)*

Detecting and characterizing signals arriving at a sensor array is a problem of practical importance in aerospace, biomedical, geological, and sonar signal processing. If the simplifying assumption of narrowband distinct signal sources can be made, the so-called *high-resolution techniques*, which are also known as *eigenvalue methods or signal subspace methods*, may be applied and offer a promise of complete and unambiguous assessment of the environment.

One of the impediments to practical application of these concepts has been implicit requirement for well-known noise structures and precise array calibration. Herein we introduce a class of techniques termed *subspace stability methods* which relax those restrictions by exploiting the temporal stability of the signal subspace. These are demonstrated to effectively process sonar array data against which conventional subspace processing fails. The most promising variation is the *Subspace Stability Exploitation Tracker (SSET)* which couples signal subspace DOA estimation algorithms with multiple target tracking techniques for accurate signal enumeration and characterization.

A novel proof of the validity of spatial smoothing to permit processing of coherent wavefronts is offered using a vector subspace perspective. This viewpoint then suggests new algorithms for coherent signal processing.

Finally, in addition to developing an array calibration algorithm amenable to on-line processing, prominent array signal processing techniques are described in a tutorial fashion and practical aspects of their performance and implementation discussed.

14. SUBJECT TERMS
frequency estimation array processing, subspace methods, subspace stability exploitation, high-resolution array processing, MUSIC, ESPRIT, GEESE, SSET

15. NUMBER OF PAGES
186

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT
UNCLASSIFIED

18. SECURITY CLASSIFICATION OF THIS PAGE
UNCLASSIFIED

19. SECURITY CLASSIFICATION OF ABSTRACT
UNCLASSIFIED

20. LIMITATION OF ABSTRACT

Abstract

Detecting and characterizing signals arriving at a sensor array is a problem of practical importance in aerospace, biomedical, geological, and sonar signal processing. If the simplifying assumption of narrowband distinct signal sources can be made, the so-called *high-resolution techniques*, which are also known as *eigenvalue methods* or *signal subspace methods*, may be applied and offer a promise of complete and unambiguous assessment of the environment.

One of the impediments to practical application of these concepts has been an implicit requirement for well-known noise structures and precise array calibration. Herein we introduce a class of techniques termed *subspace stability methods* which relax those restrictions by exploiting the temporal stability of the signal subspace. These are demonstrated to effectively process sonar array data against which conventional subspace processing fails. The most promising variation is the *Subspace Stability Exploitation Tracker* (SSET) which couples signal subspace DOA estimation algorithms with multiple target tracking techniques for accurate signal enumeration and characterization.

A novel proof of the validity of spatial smoothing to permit processing of coherent wavefronts is offered using a vector subspace perspective. This viewpoint then suggests new algorithms for coherent signal processing.

Finally, in addition to developing an array calibration algorithm amenable to on-line processing, prominent array signal processing techniques are described in a tutorial fashion and practical aspects of their performance and implementation discussed.

Table of Contents

List of Figures

List of Tables

1. Introduction

Sensor arrays are generally used for DCL—Detection, Classification, and Localization—of signals arriving at the array. This capability is important in a variety of disciplines: including aerospace, biomedical, geophysical, and sonar signal processing. If the signal sources satisfy some simple assumptions, it is possible to identify and characterize signals from the sampled sensor array covariance matrix. Since the covariance matrix is a *summary statistic*, detecting and characterizing the signals from this summary is often far more tractable from a computational perspective than working with the entire time history of the data—*assuming* that a *valid* signal and noise model is used by the DCL processing.[1]

This document is motivated by an interest in practical real-time algorithms for array signal processing. As such, prominent signal enumeration and direction-of-arrival (DOA) estimation algorithms are reviewed and practical aspects of their performance and implementation discussed. Unfortunately, the fundamental assumptions of much of the published literature are often invalid for real-world signal and array environments. To address this problem, we introduce a new class of techniques termed *subspace stability methods* which exploit the temporal stability of the source-array geometry. These are demonstrated to effectively process sonar array data against which conventional array processing techniques fail.

Other original contributions include a new proof of the validity of spatial smoothing to permit processing of coherent signal waveforms. The perspective adopted in the proof suggests new algorithms for array processing in a coherent signal environment. Additionally, a novel on-line array calibration algorithm is proposed which complements the subspace stability methods.

The sequence of this chapter introduces the array signal processing framework and explains the structure of the rest of the document.

We should also note the duality between the spatial frequency processing of array signal processing and the spectral frequency processing of a time series. As a result, the techniques discussed herein may be directly applied to time-series analysis and frequency estimation.

[1] Typical simplifying assumptions which are often violated include: uncorrelated signals, known sensor array noise covariance structure, cisoid ("narrowband") signals, perfectly known array geometries, and sensors which do not introduce any errors during the sampling.

1.1 Taxonomy and Overview

The majority of the recently developed DOA (Direction of Arrival) estimation algorithms belong to a class of techniques known variously as "eigenvalue methods", "high-resolution direction finding", "signal subspace fitting", or permutations, thereof and are geared towards the problem of detecting and locating *multiple* simultaneously impinging signals. In the literature, specific variations on these techniques are also identified by the acronyms: MUSIC (MUltiple SIgnal Characterization), MD-MUSIC (Multiple Dimensional MUSIC), root-MUSIC, ESPRIT (Estimation of Signal Parameters via Rotational Invariance Techniques), TLS-ESPRIT (Total Least Squares ESPRIT), PRO-ESPRIT (Procrustes ROtations based ESPRIT), GEESE (GEneralized Eigenvalues utilizing Signal Subspace Eigenvectors), etc. This monograph is intended as a tutorial summary of *some* of the algorithms as well as some of the supporting techniques for signal detection and enumeration or processing of coherent signals. Additionally, some novel techniques and perspectives are introduced. To illustrate the subtlety involved—as well as the academic penchant for unique acronyms and algorithms—consider the (incomplete) list of DOA estimation in the taxonomy[2] of Table 1-1

Obviously, a synopsis of *all* of the possible variations on a theme would be infeasible. Instead, we will examine the DOA estimation problem and provide a framework within which the above techniques may be described. Additionally, we shall try to develop a heuristic perspective on these techniques as well as their strengths, deficiencies, limitations, and implementation issues. We will also explore the basic assumptions with respect to signal characteristics, noise characteristics, and array geometries.

The **direct mapping** approaches (Chapter 3) are geared towards detecting and localizing a single *dominant* signal. Although sometimes implemented in the time domain, the same results may be derived directly from the (complex-valued) array covariance matrix using only Schur products and summations. Their computational efficiency makes these methods attractive despite their being biased estimators and not very resistant to multiple signal sources.

The **inverse mapping** methods (also discussed in Chapter 3) typified by the maximum entropy (ME) and minimum variance (MV) methods presume that the impinging signals have a consistent behavior as they propagate across the array—so that a propagation model is

[2] We could alternately group these algorithms by numerical approach (using Li's [55] categories): *extrema-searching* (e.g., MUSIC and Min-Norm), *polynomial rooting* (e.g., Pisarenko, Min-Norm, and Root-MUSIC), and *matrix-shifting* (e.g., state-space realization, ESPRIT, GEESE, and matrix-pencil methods). These terms as well as those of the listed taxonomy will be addressed as part of the subsequent discussions.

Table 1-1: DOA Processing Taxonomy

Direct-Mapping Methods	• Beamforming • Split-Aperture • Full-Aperture
Inverse-Mapping Methods	• Maximum Entropy (Burg) • Minimum Variance (Capon, Borgiotti & Kaplan)
Orthogonal Subspace	• Pisarenko's method • MUSIC • root-MUSIC • min-norm MUSIC, • sequential MUSIC • IES-MUSIC (ImprovEd Sequential MUSIC) • beam-space MUSIC
Signal Subspace	• ESPRIT • TLS-ESPRIT • PRO-ESPRIT • GEESE • Matrix-Pencil • State Space Realization—TAM (Toeplitz Approximation Method) • WSF (Weighted Subspace Fitting) • ML (Maximum Likelihood) • SMOR (Stochastic Model Order Reduction) • beamspace ESPRIT • MODE
Signal Detection and Enumeration	• Bartlett-Lawley • Akaike Information Criterion (AIC) • Minimum Description Length (MDL)
Subspace Stability Methods	• Burkhardt's Method • root-Tracker • SSET (Subspace Stability Exploitation Tracker)
Related Topics	• Spatial Smoothing (Coherent Signals) • Noise Modelling • Array Calibration • Broadband Signals • Dynamical Systems Approach

appropriate. Since the model fitting's extrema search uses the *inverse* of the covariance matrix to determine the DOAs, they are computationally more demanding than the direct mapping approaches. While they do provide more resolution than the classical beamforming, they are still *not* unbiased estimators of the DOA nor do they fully exploit the structure of the problem.

Schmidt [1] is generally credited as the father of "subspace processing" as well as the MUSIC algorithm; however, Pisarenko [56], Berni [40], and Bienvenu and Kopp [56] did related and supporting work. The basic idea is that a (narrowband) signal at each element of an array may be viewed as a phase-shifted replica of that available at the other array elements so that a wavefront contribution to a the receiving array elements at any instant may be constructed as[3]

$$\boldsymbol{x}(t) = \boldsymbol{d}(\theta)s(t). \tag{1-1}$$

where $s(t)$ is the waveform at a reference element, $\boldsymbol{x}(t)$ is a vector of the wavefront contribution at the array elements, and $\boldsymbol{d}(\theta)$ is a geometry-dependent vector which maps the reference signal onto the array elements. Note that the ability to separate the geometric and temporal terms in Eqn (1-1) is due to the narrowband (cisoid) signal approximation. In this case, the array covariance matrix, $\boldsymbol{R}$,

$$\boldsymbol{R} = \boldsymbol{x}\boldsymbol{x}^{\dagger} \tag{1-2}$$

is simply a *projection matrix* multiplied by a scalar. Denoting the projection due to a signal arriving from a direction θ as

$$\boldsymbol{P}_{\theta} = \boldsymbol{d}(\theta)\boldsymbol{d}(\theta)^{\dagger}\ , \tag{1-3}$$

we see that $\boldsymbol{P}_{\theta}$ has a rank of 1 since each row (or column) is just a scaled (by a complex-value) replica of the other columns. Thus, just as a straight line in Cartesian space only has one dimension[4], the *signal subspace* is of lower rank than the array size.

Of course, in a real system the sampled covariance matrix, $\hat{\boldsymbol{R}}$ will be corrupted by noise so that $\hat{\boldsymbol{R}} \neq \boldsymbol{R}$ and the covariance matrix will be of full rank—i.e., corresponding to the number of sensors rather than the number of signals. Schmidt realized that an eigendecomposition of $\hat{\boldsymbol{R}}$

[3] A general assumption in the DOA estimation problem is to assume that the signals are *analytic*—i.e., complex-valued. Since any real-valued time series may be expressed in analytic components via Euler's identity, this does not impose any practical implementation restrictions—other than computational loading.

[4] We could chose a coordinate system "aligned" with the line so that one axis was the "line subspace" and all the other coordinates would be the "non-line subspace".

could be used to identify the signal subspace since "large" (high power) eigenvalues and vectors would correspond to the signal subspace. Since the non-signal eigenvectors constituted an **orthogonal subspace**, hypothesized DOAs [i.e., $\boldsymbol{d}(\theta)$] could be projected onto the noise subspace, with the signal direction(s) identified by minima in the projection. Algorithms exploiting the orthogonality of the signal and noise subspaces are discussed in Chapter 5.

One problem with the orthogonal subspace approach is that a search for the minima of the projection of the hypothesized directions onto the noise subspace tends to be computationally expensive—especially if there are few signals relative to the size of the array—as well as the normal problems associated with searching for a global maxima and resolving adjacent maxima. Roy and Kailath [9] suggested an approach (ESPRIT) for working directly with the **signal subspace**—which was a path soon followed by many others. These algorithms generally require a linear, uniformly spaced array as opposed to the arbitrary topologies possible with MUSIC. However, this additional covariance matrix structure permits *much* more efficient implementations. Although the ESPRIT algorithm is sensitive to noise-induced perturbations, subsequent algorithms (e.g., GEESE, TLS-ESPRIT, etc.) are much more robust. Chapter 6 addresses the signal subspace algorithms in more detail.

The optimal[5] array processor integrates the signal detection and DOA estimation process [1]; however, to avoid the computational load associated with this implicitly multi-dimensional optimization problem, the subspace array processing is typically implemented as a two stage process with the DOA estimation following an initial determination of the number of signals derived from an analysis of the sampled array covariance matrix eigenvalues[6]. The DOA algorithms are typically sensitive to an accurate estimate of the number of signals—so that the real limitation of subspace methods is imposed by the **signal enumeration** problem rather than the ability to estimate the source directions; in essence, if the number of signals can be accurately determined, the associated directions can also be accurately determined. To avoid the problems of *ad hoc* threshold definition by experts based upon data analysis, *information theoretic criterion* were developed which attempt to enumerate the signals based upon the sampled covariance matrix in conjunction with knowledge of sampling intervals and noise bandwidths. Although a variety of detectors have been developed using these concepts (see

[5] "Optimal" here is defined in the scientific sense as opposed to the engineering sense which incorporates the somewhat nebulous aspects of feasibility and implementation difficulty as part of the cost function used to define "optimal."

[6] Such an eigenvalue analysis does not exploit *a priori* knowledge of the array topology and is, therefore, suboptimal in the scientific sense.

Chapter 4), they are *very* sensitive to the underlying assumptions of perfect sensors and perfect noise and, as a result, generally do not work very well when applied to acoustic data.

Motivated by a desire for subspace processing techniques which were suitable for on-line real-time data processing and recognizing the fundamental limitations of the existing signal enumeration approaches, we have identified a novel class of algorithms which exploit the temporal stability of the source-array geometry. These **Subspace Stability Methods** (Chapter 7) build upon concepts developed by Burkhardt [59] who realized that averaging the spatial spectra resulting from Pisarenko's method eventually suppressed spurious noise-induced spectral peaks; this is analogous to spectral averaging of time-series data. Several algorithms are proposed and explored in addition to *Burkhardt's Method.* One of the most promising is to combine the GEESE signal subspace DOA estimation algorithm with multiple hypothesis enumeration and multiple target tracking (MTT) techniques; the resulting *Subspace Stability Exploitation Tracker* (SSET) is demonstrated to accurately enumerate, identify, and locate signal sources contained within acoustic data. Other algorithms include the *root-Tracker* which uses a constant enumeration hypothesis coupled with MTT concepts. A root-Tracker implementation is developed based upon an AR implementation of Pisarenko's method

The subspace processing approach contains implicit assumptions about the signals, noise, and sensors. While techniques are available to mitigate the effects of violating some of these assumptions, these techniques add algorithmic complexity as well as computational load to the signal processing. Chapter 8 reviews the implications of mobile sources transmitting non-cisoid[7] signals to non-ideal sensor arrays in the presence of non-ideal noise while Chapter 9 addresses the use of spatial smoothing to preprocess sampled covariance matrices to facilitate detection and DOA estimation of coherent and correlated signals. A new proof of the validity of spatial smoothing is presented which adopts a vector subspace perspective rather than the conventional matrix algebra approach. This viewpoint facilitates some new approaches for coherent signal processing which are also discussed.

Even assuming the idealized assumptions of the signal, noise, and sensor hold, the sampled covariance matrix will differ from the asymptotic form due to the non-infinite sampling interval. Appendix A presents the statistics of the sampled covariance matrix elements as a function of signal and noise bandwidth and sampling interval.

[7] A *cisoid* is a complex sinusoid, i.e., $Ae^{\iota(\omega t + \psi)}$.

1.2 Basic Assumptions and Constraints

There are some basic assumptions about the signals and array geometry which are used in the discussed techniques:

- fewer signals than array elements;
- linear, uniformly spaced array;
- narrowband signals;
- far-field sources (impinging *plane* waves);
- less than half-wavelength element spacing;
- accurate and calibrated sensors;
- known sensor noise covariance structure; and,
- accurate signal enumeration.

Although some algorithms (e.g., MUSIC) can handle arbitrary array topologies, linear uniformly spaced array should be presumed as required unless otherwise stated. Reviewing the above list, note that a number of real-life situations which could potentially violate these basic ground rules; for example, towed arrays could possible not be linear (due to “snaking” or sensor misplacement) or a phase or gain bias might be introduced by the sensors. Furthermore, if the array was “large” the narrowband assumption might start to break down such that the time delay at the extreme elements cannot be modelled by just a simple phase shift of the impinging signals. Additionally, there are some conditions which simplify the algorithms:

- spatially independent equi-powered Gaussian noise
- uncorrelated signals

Even though coherent signals and correlated noise can be handled through spatial smoothing and estimation of the noise covariance structure, additional complexity—and computational loading—is required. Spatially correlated noise might be derived from flow noise or common electronic circuitry while correlated signals could arise either unintentionally through multipath effects or be due to similar signals from other sources.

1.3 Definition of Variables

This section summarizes the various variables and notations used in this paper for the convenience of the reader. In general, lower-case boldface (e.g., $\boldsymbol{v}$) represents vectors, boldfaced upper-case (e.g., $\boldsymbol{B}$) indicate matrices, upper-case italic (K) indicates constants, and lower-case italic (n) indicates scalars. The operator $(\cdot)^{\dagger}$ indicates the Hermitian (complex-conjugate transpose) while $(\cdot)^{*}$ is used to denote the conjugate and $(\cdot)^{\mathrm{T}}$ the transpose. The variables are summarized in Table 1-2.

Table 1-2: Definition of Variables and Notation

J	the number of impinging wavefronts/signals,
K	the number of staves/elements in the receive array,
N	the number of samples taken at a sampling rate of f_S
s_j	the j^{th} *signal* arriving at the array.
$\boldsymbol{R}$	the *expected* covariance matrix (including both noise and signal contributions).
$\hat{\boldsymbol{R}}$	the *sampled* covariance matrix (including both noise and signal contributions).
$\boldsymbol{R}_s$	the signal covariance matrix. (Rank ≤ J where the equality holds if the signals are at most partially coherent.)
τ_j	the stave-to-stave propagation time of the j^{th} signal wavefront.
$\boldsymbol{s}(t)$	vector of the J arriving waveforms
$\boldsymbol{D}$	a "steering matrix" which maps the J wavefronts onto the K receive elements,
$\boldsymbol{d}_j$	the "steering vector" mapping the j^{th} signal onto the K receive elements.
Σ_n	the noise covariance matrix (equal to $\sigma^2\boldsymbol{I}$ for *i.i.d.* spatially uncorrelated noise),
λ_k	the k^{th} eigenvalue of the sampled covariance matrix,
$\boldsymbol{b}_k$	the eigenvector associated with λ_k,
$\boldsymbol{\Phi}$	stave transition matrix characterizing adjacent element phase shifts for the incoming signals,
M	the number of staves/elements in the spatial smoothing subarrays,

Table 1-2: Definition of Variables and Notation (Continued)

L	the number of forward subarrays of M elements/staves in spatial smoothing
δ	inter-element array spacing
θ_j	the DOA of the j^{th} signal
φ_j	the inter-element phase shift of the j^{th} signal = $\frac{2\pi f}{c}\sin\theta_j$

2. The Sampled Covariance Matrix

This chapter explores the assumed structure of the sampled signal covariance matrix which is exploited by all of the subsequent techniques. In the discussion, some of the notation and assumptions implicit in the subsequent detection and localization algorithms are established.

The chapter concludes with a discussion of the signal and noise subspaces of the covariance matrix and provides an intuitive perspective of the distinguishing characteristics of each subspace and how those characteristics may be exploited in the signal detection and localization problems.

2.1 Introduction

The approaches discussed in the ensuing chapters presume J narrowband planar wavefronts impinging upon a K-element array so that the received signal vector, $\boldsymbol{x}(t)$, may be modelled as

$$\boldsymbol{x}(t) = \boldsymbol{D}(\theta)\boldsymbol{s}(t) + \boldsymbol{n}(t), \tag{2-1}$$

where $\boldsymbol{n}(t)$ is additive noise, $\boldsymbol{s}(t)$ is a the vector of impinging wavefronts at some reference location, and $\boldsymbol{D}(\theta)$ is the *steering matrix* which maps the wavefronts onto the elements of the array—where the steering matrix is a function only of the array-signal geometry. Our problem is to detect signals in the received signal and to extract the associated geometry. It is often possible—and computationally expedient—to extract the desired information from the "summary statistic" known as the sampled covariance matrix,

$$\hat{\boldsymbol{R}} = \overline{\boldsymbol{x}\boldsymbol{x}^{\dagger}} = \overline{(\boldsymbol{D}\boldsymbol{s}+\boldsymbol{n})\;(\boldsymbol{D}\boldsymbol{s}+\boldsymbol{n})^{\dagger}} = \frac{1}{T}\int_{t_0}^{(t_0+T)} \boldsymbol{x}\boldsymbol{x}^{\dagger}\,dt \tag{2-2}$$

where the final form presumes that the covariance matrix has been derived from the signal received during an interval of T and the $\bullet^{\dagger}$ denotes the Hermitian transpose (complex-conjugate transpose). In the following sections we will explore this data model in more detail as well as the structure of the resulting covariance. Appendix A addresses the statistics of the covariance matrix due to varying signal and noise models; for the purposes of this chapter, we will presume sufficient integration times so that the sampled covariance matrix has effectively converged to that of the idealized matrix—i.e., subspace processing is appropriate.

2.2 Basic Assumptions and Constraints

Although the MUSIC algorithm does not require a uniform linear array or far-field sources, most of the other DOA estimation algorithms require such a configuration; hence, herein we shall assume that the arrays have a common axis and have a constant inter-element spacing. To reiterate Section 1.2, the fundamental as well as the simplifying assumptions of subspace processing are listed in Table 2-1.

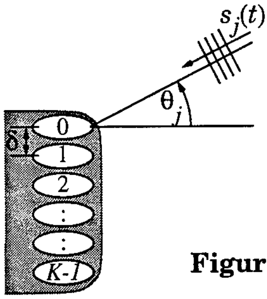
Table 2-1: Fundamental and Simplifying Assumptions

Fundamental Assumptions	• fewer signals than array elements • linear, uniformly spaced array • narrowband signals • constant source-array geometries • far-field sources (impinging *plane* waves) • less than half-wavelength element spacing • an accurate estimate of the number of signals • accurate and calibrated sensors
Simplifying Assumptions	• spatially independent equi-powered Gaussian noise • uncorrelated signals

An additional constraint—which is also a motivation for the subspace algorithms, in general—is the need for computational efficiency and accurate and reliable detection, classification, and localization (DCL). These concerns are addressed as part of the algorithmic development and analysis.

2.3 The Impinging Signal Model

Assume a K-sensor array having a constant (center-to-center) sensor spacing of δ. Furthermore, assume that impinging on those sensors are J narrowband plane waves such that the propagation time across the array is small relative to the temporal variability of the signal amplitude and phase modulation (i.e., the signal variability is "slow" relative to the propagation across the array); this permits us to view the signal at any sensor at a given



"0" indicates either the left or top sensor.

Figure 2-1: Signal and Array Geometry

sampling epoch as a phase-shifted version of that received at the other sensors. The wavefront from the j^{th} signal arriving at the k^{th} sensor may be expressed as

$$s_{jk}(t) = A_j(t-k\tau_j)e^{\iota\psi_j(t-k\tau_j)} \cong A_j(t)e^{\iota(\psi_j(t)-k\varphi_j)}, \tag{2-3}$$

where τ_j is the signal geometry-dependent propagation time from one sensor to an adjacent sensor,

$$\tau_j = \frac{\delta\sin\theta_j}{c}. \tag{2-4}$$

The geometry and sensor numbering are defined as shown in Figure 2-1 and c is the wavefront propagation speed. The sensor-to-sensor phase shift, φ_j, in Eqn (2-3) is a signal geometry and frequency dependent term defined as,

$$\varphi_j \cong \frac{d}{dt}\psi_j(t)\tau_j = \omega_j(t)\tau_j, \tag{2-5}$$

where the approximation is valid due to our previous assumption about a slowly varying signal (in this case, signal frequency) relative to the propagation across the array. Obviously, if the impinging signal is cisoidal, the frequency $\omega_j(t)$ could be treated as (approximately) a constant. In general, the expected phase shift is,

$$E(\varphi_j) = \overline{\varphi_j} = \frac{\delta\overline{\omega}_j\sin\theta_j}{c}, \tag{2-6}$$

where $\overline{\omega}_j$ is the mean signal frequency. Expressing the sensor spacing in terms of wavelengths, since $\lambda = c/f$, leads to

$$E(\varphi_j) = \overline{\varphi_j} = 2\pi\overline{\delta_{\lambda_j}}\sin\theta_j. \tag{2-7}$$

In the following discussion, we generally assume that the signal may either be treated as cisoidal or that we will be averaging over a sufficient duration so that the expectation of Eqn (2-6) may be used. As an aside, note that satisfying the narrowband assumption may be accomplished for wideband signals by applying an FFT to the data and treating each resulting frequency bin as a separate signal—with the caveat that the data set be long relative to the propagation time across the array.

We can view the j^{th} signal's contribution at the k^{th} sensor as a geometry-dependent modification of the signal received at some reference location. Thus, Eqn (2-3) could be viewed as,

$$s_{jk}(t) = (A_j(t)e^{\imath\psi_j(t)})e^{-\imath k\varphi_j} = s_j(t)e^{-\imath k\varphi_j} = d_k(\theta_j)s_j(t), \qquad (2\text{-}8)$$

where $s_j(t)$ is the signal received at the reference point (i.e., sensor 0) and $d_k(\theta_j)$ is the phase shift implicitly defined by Eqn (2-8). In general, the sampled signal x_k will contain contributions from a number of signals as well as sensor-specific noise; hence, the signal at any element may be expressed as,

$$x_k(t) = \sum_{j=1}^{J} s_{jk}(t) + \sigma_k = \sum_{j=1}^{J} d_k(\theta_j)s_j(t) + n_k(t). \qquad (2\text{-}9)$$

Extending this argument and notation, the sampled signal vector may be represented as

$$\boldsymbol{x} = \boldsymbol{D}\boldsymbol{s} + \boldsymbol{n}, \qquad (2\text{-}10)$$

where $\boldsymbol{x}$ is the vector of the signal received at each sensor,

$$\boldsymbol{x} = \begin{bmatrix} x_0(t) \\ x_1(t) \\ \vdots \\ x_{K-1}(t) \end{bmatrix}. \qquad (2\text{-}11)$$

$\boldsymbol{D}$ is the *steering matrix*,

$$\boldsymbol{D} = \left[\boldsymbol{d}_1\ \boldsymbol{d}_2\ \dots\ \boldsymbol{d}_j\ \dots\ \boldsymbol{d}_J\right] = \begin{bmatrix} d_0(\theta_1) & d_0(\theta_2) & \dots & \dots & d_0(\theta_J) \\ d_1(\theta_1) & d_1(\theta_2) & \dots & & d_1(\theta_J) \\ d_2(\theta_1) & & & & d_2(\theta_J) \\ \vdots & & & & \vdots \\ d_{K-1}(\theta_1) & \dots & d_{K-1}(\theta_j) & \dots & d_{K-1}(\theta_J) \end{bmatrix}, \tag{2-12}$$

where the *steering vector*, $\boldsymbol{d}_j$, in Eqn (2-12) maps the $s_j(t)$ signal onto the receive array. This vector is implicitly defined for a uniform linear array as,

$$\boldsymbol{d}_j = \begin{bmatrix} d_0(\theta_j) \\ d_1(\theta_j) \\ d_2(\theta_j) \\ \dots \\ d_{K-1}(\theta_j) \end{bmatrix} = \begin{bmatrix} 1 \\ e^{-\iota\varphi_j} \\ e^{-\iota 2\varphi_j} \\ \vdots \\ e^{-\iota k\varphi_j} \\ \vdots \\ e^{-\iota(K-1)\varphi_j} \end{bmatrix} = \begin{bmatrix} 1 \\ \nu_j \\ \nu_j^2 \\ \vdots \\ \nu_j^{K-1} \end{bmatrix}. \tag{2-13}$$

where we recall that the sensor-to-sensor phase shift increment, φ_j, is

$$\varphi_j = \frac{\delta \sin\theta_j}{c}\omega_j(t). \tag{2-14}$$

Exploiting the narrowband assumption so that inter-element spacing may be expressed in units of wavelengths, δ_λ, leads to the sensor-to-sensor phase shift increment of,

$$\varphi_j = 2\pi\delta_\lambda \sin\theta_j. \tag{2-15}$$

Furthermore, the impinging signal vector is defined as,

$$\boldsymbol{s} = \boldsymbol{s}(t) = \begin{bmatrix} s_1(t) \\ s_2(t) \\ s_3(t) \\ \vdots \\ s_J(t) \end{bmatrix}, \tag{2-16}$$

with $\boldsymbol{n}$ being the additive noise vector,

$$\boldsymbol{n} = \boldsymbol{n}(t) = \begin{bmatrix} n_0(t) \\ n_1(t) \\ \vdots \\ n_K(t) \end{bmatrix}. \qquad (2\text{-}17)$$

Rather than working directly with the $K \times N$ sampled data matrix (where N is the number of sampling epochs), for computational efficiency, the subspace processing algorithms generally use the $K \times K$ array covariance matrix.

2.4 The Sampled Signal Covariance Matrix

The array covariance matrix, $\boldsymbol{R}$ of the K sensor array is the $K \times K$ matrix of the expected value of the outer product of the received signal with its complex-conjugate,

$$\boldsymbol{R} = E\{\boldsymbol{x}\boldsymbol{x}^{\dagger}\}, \qquad (2\text{-}18)$$

where $E\{\cdot\}$ is the statistical expectation. Obviously, information loss occurs if the representation of Eqn (2-18) which contains $K(K-1)/2$ independent elements is used instead of the much larger KN samples in the sampled data set. The use of a received energy rather than waveform representation limits the array resolution to a maximum of $K-1$ signals[1] as well as increasing the required dynamic range[2]. Using the definition for $\boldsymbol{x}$ of Eqn (2-10) in Eqn (2-18) yields,

$$\boldsymbol{R} = E\{(\boldsymbol{D}\boldsymbol{s}+\boldsymbol{n})(\boldsymbol{D}\boldsymbol{s}+\boldsymbol{n})^{\dagger}\}, \qquad (2\text{-}19)$$

or, equivalently,

$$\boldsymbol{R} = E\{\boldsymbol{D}\boldsymbol{s}\boldsymbol{s}^{\dagger}\boldsymbol{D}^{\dagger} + \boldsymbol{D}\boldsymbol{s}\boldsymbol{n}^{\dagger} + \boldsymbol{n}\boldsymbol{s}^{\dagger}\boldsymbol{D}^{\dagger} + \boldsymbol{n}\boldsymbol{n}^{\dagger}\}. \qquad (2\text{-}20)$$

The convergence of Eqn (2-20) is discussed in more detail in Appendix A; however, the following sections briefly review the asymptotic limits.

[1] However, to resolve more than $K-1$ using the raw data implies a valid waveform model so this is not generally a concern.

[2] Using the signal power rather than amplitude may pose implementation concerns.

2.4.1 Noise-Signal Covariance Contribution

Assuming that the signals and noise are uncorrelated and the noise has zero mean, then the cross-product matrices converge to zero, i.e.,

$$E[\boldsymbol{s}\boldsymbol{n}^{\dagger}] = 0 \qquad E[\boldsymbol{n}\boldsymbol{s}^{\dagger}] = 0. \tag{2-21}$$

2.4.2 Noise Covariance Contribution

Assuming the noise has zero mean and is *uncorrelated from element to element* while having identical statistics (i.e., power levels and spectra) leads to the noise covariance matrix, Σ_n, representation as

$$\Sigma_n \equiv E[\boldsymbol{n}\boldsymbol{n}^{\dagger}] = \begin{bmatrix} \sigma^2 & 0 & \dots & 0 & 0 \\ 0 & \sigma^2 & 0 & : & 0 \\ : & 0 & & & : \\ 0 & \dots & 0 & \sigma^2 & 0 \\ 0 & 0 & \dots & 0 & \sigma^2 \end{bmatrix} = \sigma^2 \boldsymbol{I}_K, \tag{2-22}$$

where $\boldsymbol{I}_K$ is the $K \times K$ identity matrix and σ^2 is the expected noise power. Some array environments (e.g., acoustical arrays dominated by flow noise) will not have the idealized noise covariance structure indicated by Eqn (2-22); if the general structure is known, the sampled covariance matrix may be whitened so that the noise covariance contribution has this idealized structure. The diagonal noise structure of Eqn (2-22) as two significant consequences:

- eigendecomposition naturally partitions the sampled covariance into signal and orthogonal subspaces; and,
- the information-theoretic enumeration criteria of Chapter 4 are applicable.

Although an accurate partitioning is essential for application of subspace processing techniques, the eigendecomposition-based partitioning may still be valid given “strong” signals. If the information-theoretic criteria are not valid, accurate enumeration and DOA estimation *may* be achieved through alternate approaches such as the SSET algorithm proposed in Section 7.6. In general, *some* assumption concerning the noise structure *must* be made—otherwise, *any* sampled covariance matrix could be rationalized as due to additive noise.

2.4.3 Signal Covariance Contribution

Turning our attention to the signal contribution to the covariance matrix, note that for time spans such that $\boldsymbol{D}$ is approximately constant (thereby implicitly imposing a restriction on the source-array relative motion during the averaging interval) and continuing with the assumption of narrowband signals, the expectation becomes

$$E[\boldsymbol{D}\boldsymbol{s}\boldsymbol{s}^{\dagger}\boldsymbol{D}^{\dagger}] = \boldsymbol{D}E[\boldsymbol{s}\boldsymbol{s}^{\dagger}]\boldsymbol{D}^{\dagger} \tag{2-23}$$

If the signals are uncorrelated, then the signal covariance matrix has the very simple form,

$$\boldsymbol{R}_s = E[\boldsymbol{s}(t)\boldsymbol{s}^{\dagger}(t)] \cong \begin{bmatrix} A_1^2 & 0 & 0 & \dots & 0 \\ 0 & A_2^2 & 0 & \dots & 0 \\ 0 & 0 & A_3^2 & \dots & : \\ : & : & : & & 0 \\ 0 & 0 & \dots & 0 & A_J^2 \end{bmatrix} . \tag{2-24}$$

where A_j is the signal amplitude implicitly defined by Eqn (2-3). In either case, the signal contribution to the sampled covariance matrix may be represented as,

$$\boldsymbol{R} = \boldsymbol{D}\boldsymbol{R}_s\boldsymbol{D}^{\dagger} , \tag{2-25}$$

Consistent with our assumptions, $\boldsymbol{R}_s$ is a $J \times J$ matrix and is, in general, smaller than the rank of $\boldsymbol{R}$—i.e., there are fewer signals than there are sensors in the array.

Note that the form of Eqn (2-24) will be invalid if the incoming signals are correlated—which implies that $\boldsymbol{R}_s$ is not of full (J) rank. Such a situation could occur due to multipath propagation of signals or signal repeaters. However, if the multipath introduces a significantly different Doppler shift, the signals might be decorrelated. The signals will also effectively be decorrelated if the signals are random and sufficient propagation delay offset is present so that the covariance calculation perceives them as being different signals.

If the waveforms are correlated, preprocessing the array via spatial smoothing will increase the rank of the signal covariance contribution to correspond to the number of wavefronts rather than the number of signals. Spatial smoothing is discussed in detail in Chapter 9 wherein it is shown that the array processing may be applied transparent to whether spatial smoothing was or was not used.

2.4.4 The Sampled Covariance Matrix

Since the sampled data stream contains both signals and noise, a method is required to determine the number of signal sources and their directions. If we view the sampled signal vector as

$$\boldsymbol{x}(t) = \sum_{j=1}^{J} \boldsymbol{d}_j(\varphi_j) s_j(t) + \boldsymbol{n}(t), \tag{2-26}$$

then, in principle, we seek the *non-orthogonal* basis set, $\boldsymbol{D}$, of rank J which best fits the sampled covariance matrix, $\hat{\boldsymbol{R}}$, i.e., to determine the coefficients, $\hat{\vec{\theta}}$, $\boldsymbol{R}_s$, and σ^2 which minimizes,

$$Q(J, \hat{\vec{\theta}}, \sigma^2, \boldsymbol{R}_s) = \left\| \boldsymbol{D}\boldsymbol{R}_s\boldsymbol{D}^{\dagger} + \sigma^2\boldsymbol{I} - \hat{\boldsymbol{R}} \right\| . \tag{2-27}$$

Of course, the basis set is constrained by the array topology. Such a multi-dimensional optimization approach is computationally intensive and generally impractical—albeit optimal from an accuracy perspective. Thus, the conventional approach is to adopt a two-stage process: the first stage involves signal detection based upon an eigen*value* analysis while the second stage uses the detection results along with the eigen*vectors* to determine the DOAs which best fit the sampled data.

The conventional enumeration and DOA processing algorithms implicitly assume the idealized covariance matrix structure of Eqn (2-25). Reality, however, is that due to finite sampling intervals and violations of the underlying assumptions, the sampled covariance matrix will not achieve the ideal form—with an associated penalty in enumeration performance and DOA accuracy. Much of the subspace processing literature is devoted to deriving asymptotic expressions of algorithmic performance under *very* limited scenarios. Although these analyses are mathematically aesthetic, from a practical perspective the sampled covariance matrix will be derived from data sampled over an interval and the detection and performance will *have* to be acceptable since other options either do not exist or are too computationally demanding. To illustrate, the choice of averaging interval must trade-off selection of a long interval, which increases the effective signal contribution, versus the selection of a short interval which reduces the potential for errors due to dynamic source-array geometries.

3. Classical DOA Methods

As discussed in Chapter 2, the sampled covariance matrix contains sufficient information to detect and locate impinging signals. The subspace-based techniques discussed in the following chapters represent an advance in the sense that multiple signals may be *simultaneously* detected and located. This advantage is achieved at a cost of increased algorithmic complexity—which implies correspondingly, increased computational demands. If computational loading is a design issue or if *a priori* knowledge indicates that only a single *dominant* signal source is of interest, the "classical" methods discussed in this chapter may be an appropriate choice.

3.1 Introduction and Taxonomy

Classical DOA estimation—although not employing a subspace decomposition—also utilizes the sampled covariance matrix. As illustrated in the table below, the classical methods may be divided into the direct-mapping methods which operate directly on the sampled covariance matrix and the inverse-mapping methods which utilize the inverse of the sampled covariance matrix.

	One Signal	*Multiple Signals*
Direct-Mapping	*Split-Aperture* *Full-Aperture* *Beamforming*	*Beamforming*
Inverse-Mapping	*Maximum Entropy (Burg)* *Minimum Variance (Capon and Borgiotti & Kaplan)*	

The primary advantage of these methods is one of implementation efficiency. If a single dominate signal source is present, the split-aperture or full-aperture methods can provide an accurate estimate of the source direction while demanding relatively little in the way of computational resources. If multiple signals are present, the beamforming, maximum entropy, or minimum variance methods can estimate the signal directions in a computationally simple manner. Due to making fewer assumptions about the signals, noise, and array, these methods tend to be more robust than the subspace methods—at the expense of sacrificing accuracy and performance when the subspace processing assumptions hold.

The *direct-mapping methods* operate directly on the covariance matrix—although simple, this works well in the case of a single strong wavefront impinging on the array. In more complex scenarios, the *inverse-mapping* methods may be employed; these utilize the inverse of the sampled covariance matrix and *are* capable of resolving multiple signals—albeit without the angular resolution of the subspace approaches.

Issues such as detection thresholds, accuracy, or implementation architectures[1] are only addressed heuristically. Books have been devoted to the array processing problem (e.g., [68] and [69]) so this chapter is oriented towards the intuitive rather than the mathematically rigorous. Additionally, for many applications the simplistic assumptions concerning sensor performance and noise distributions are invalid; in this case, algorithm thresholds must be derived from sample data sets rather than inaccurate mathematical models.

3.2 Single-Source Direct-Mapping Methods

In this section we explore the structure of the sampled covariance matrix in the event of a single impinging signal as well as the full-aperture and split-aperture DOA estimation methods. Although the beamforming, minimum variance, and maximum entropy methods can also be used in the single source case, their discussion is deferred to subsequent sections.

The direct mapping methods utilize the sampled covariance matrix without the need to perform matrix inversions or decompositions.[2] For this reason, they are relatively efficient computationally—although, not as resilient to noise and multiple sources as other methods. Contrary to the normal sequencing, we will discuss the DOA estimation methods prior to addressing the direct-mapping signal detection problem. The rationale is that, for many practical applications, the signal detection is more properly viewed as a "quality indicator" issue since—due to the low computational loading—a DOA estimate is always generated. The three direct mapping DOA estimation approaches are:

- Beamforming

[1] As an aside, it should be noted that in a strong SNR environment, the sampled covariance matrix will be close to singular. This could cause numerical problems for the inverse-mapping approaches; however, if anticipated, numerical instabilities may be easily avoided.

[2] Although the split-aperture and full-aperture methods are often implemented in the time domain on the sampled data stream, such implementations are functionally and mathematically equivalent to implementations operating on the sampled covariance matrix.

- Full-Aperture
- Split-Aperture

The beamforming approach may be viewed as a search technique wherein a source DOA is hypothesized and the hypothesis correspondence with the sampled data determined. The split-aperture approach may be viewed as a root-finding technique wherein the relative sampled signal phase between two spatially separated points (phase centers) is used to determine the DOA. Unfortunately, DOA ambiguities are introduced when the phase centers are separated by more than one-half wavelength. As an alternative, Rubano [36] proposed the full-aperture method which—upon appropriate choice of weights—does not have a phase ambiguity. As will be shown, the split-aperture method may be viewed as a special case of the full-aperture approach.

3.2.1 The Structure of the Single-Source Covariance Matrix

Continuing the notation developed in Chapter 2, any given *narrowband* analytic[3] signal, $s(t)$, impinging upon the K-element array at an angle θ from the array may be mapped into the signal received at each array element by

$$\boldsymbol{x}(t) = \boldsymbol{d}(\theta)s(t) \tag{3-1}$$

where $\boldsymbol{d}(\theta)$ is the *steering vector*; this vector corresponds to the propagation-induced phase shift relative to signal at some reference location. Note that the decomposition of Eqn (3-1) into separable geometric and temporal components *requires* a narrowband signal so that the effect of propagation across the array may be legitimately modelled as a phase shift! Eqn (3-1) applies for arbitrary array topologies; if we presume that the array is a one-dimensional array with uniformly spaced elements, then if the inter-element phase shift may be expressed as

$$\varphi = \frac{2\pi f}{c}\delta\sin\theta, \tag{3-2}$$

where δ is the inter-element spacing, f is the nominal signal frequency, and c is the wavefront propagation speed. The steering vector is simply

[3] Since a real-valued signal may be relatively easily converted to its analytic equivalent, this restriction to complex-valued signals does not pose any mathematical obstacles.

$$\boldsymbol{d} = \begin{bmatrix} d_0(\theta) \\ d_1(\theta) \\ d_2(\theta) \\ \dots \\ d_{K-1}(\theta) \end{bmatrix} = \begin{bmatrix} 1 \\ e^{-\iota\varphi} \\ e^{-\iota 2\varphi} \\ \vdots \\ e^{-\iota k\varphi} \\ \vdots \\ e^{-\iota(K-1)\varphi} \end{bmatrix}. \tag{3-3}$$

As discussed in Chapter 2, the covariance matrix resulting from an impinging narrowband signal, $s(t) = A(t)e^{\iota\psi(t)} \approx Ae^{\iota\omega t}$, in a noise-free environment can be expressed as

$$\boldsymbol{R} = E[\boldsymbol{x}\boldsymbol{x}^{\dagger}] = E[\boldsymbol{d}s(t)(\boldsymbol{d}s(t))^{\dagger}] = \boldsymbol{d}E[|s(t)|^2]\boldsymbol{d}^{\dagger} = \boldsymbol{d}\boldsymbol{R}_s\boldsymbol{d}^{\dagger}. \tag{3-4}$$

For this single signal case, the signal covariance is a scalar corresponding to the average signal power, $\boldsymbol{R}_s \equiv E[|A|^2]$. Therefore, the array covariance matrix is the outer product,

$$\boldsymbol{R} = (E[|A|^2])\boldsymbol{d}\boldsymbol{d}^{\dagger} = \overline{|A^2|}\begin{bmatrix} 1 \\ e^{-\iota\varphi} \\ e^{-\iota 2\varphi} \\ e^{-\iota 3\varphi} \\ \vdots \\ e^{-\iota\varphi(K-1)} \end{bmatrix}\begin{bmatrix} 1 & e^{\iota\varphi} & e^{\iota 2\varphi} & e^{\iota 3\varphi} & \dots & e^{\iota\varphi(K-1)} \end{bmatrix} \tag{3-5}$$

or, equivalently,

$$\boldsymbol{R} = \overline{|A^2|}\begin{bmatrix} 1 & e^{\iota\varphi} & e^{\iota 2\varphi} & e^{\iota 3\varphi} & \dots & e^{\iota\varphi(K-1)} \\ e^{-\iota\varphi} & 1 & e^{\iota\varphi} & e^{\iota 2\varphi} & \dots & e^{\iota\varphi(K-2)} \\ e^{-\iota 2\varphi} & e^{-\iota\varphi} & 1 & e^{\iota\varphi} & \dots & e^{\iota\varphi(K-3)} \\ e^{-\iota 3\varphi} & e^{-\iota 2\varphi} & e^{-\iota\varphi} & 1 & \dots & e^{\iota\varphi(K-4)} \\ \vdots & \vdots & \vdots & \vdots & & \vdots \\ e^{-\iota\varphi(K-1)} & e^{-\iota\varphi(K-2)} & e^{-\iota\varphi(K-3)} & e^{-\iota\varphi(K-4)} & \dots & 1 \end{bmatrix}. \tag{3-6}$$

Under these assumptions $\boldsymbol{R}$ is *Toeplitz*[4] (the same value along diagonals of the matrix) as well as being *Hermitian* (complex-conjugate symmetric). Due to sensor errors, noise, source movement, sampling effects, etc., this idealized structure will not be achieved by the sampled covariance matrix; however, it is assumed that the perturbation from the ideal structure is negligible. If the sampled covariance matrix is not Toeplitz, this implies either the absence of a signal (assuming spatially uncorrelated noise, the covariance matrix would converge to a diagonal structure) or the presence of correlated sources. Appendix A provides a discussion of the statistics associated with the elements of the sampled covariance matrix.

The rank of the array covariance matrix (neglecting noise contributions and other perturbations) will correspond to the number of impinging wavefronts since each signal contribution to a given sensor is simply a (complex) scalar multiple of the contribution to the other sensors. In the event of a single signal and the absence of noise, the eigenvalues of the covariance matrix would be $\{KA^2, 0, \ldots, 0\}$—so the largest eigenvalue is an estimator of the signal power. In the event of multiple signals, this simple interpretation of the eigenvalues breaks down.

3.2.2 Split-Aperture DOA Estimation

Unlike beamforming (discussed in Section 3.3.1) which does a search to determine the direction with the maximum signal power, the split-aperture assumes that a *single* signal is present and determines the phase shift—which directly maps into the geometric angle—between the phase centers of two subarrays. Partitioning a K-element array into two M-element subarrays, as is illustrated in Figure 3-1 for a uniform linear array, allows us to calculate a phasor[5] via,

$$\beta e^{\iota\psi} = \int_T \left(\sum_{m=0}^{M-1} w_m x_m \right) \left(\sum_{m=0}^{M-1} w_m x_{m+m_0} \right)^* dt \qquad (3\text{-}7)$$

where w_m is an element-specific weighting which also incorporates a phase shift, $d_m(\theta)$, to electrically align the array elements as well as providing array directivity characteristics and x_k denotes the signal received at the k^{th} array element. Interchanging the order of integration and summation permits Eqn (3-7) to be expressed in matrix notation in terms of the covariance matrix,

[4] In fact, the *expected* covariance matrix will have a Toeplitz structure for multiple impinging wavefronts—*providing* the impinging signals are uncorrelated (so that the signal covariance matrix is diagonal). If multiple signals are present, the diagonals will be more complicated functions of the scenario geometry than illustrated in Eqn (3-6) with different amplitudes along the diagonals. Additive noise and sampling effects will perturb this idealized Toeplitz structure; however, by construction, the sampled covariance matrix will *always* be Hermitian.

[5] A phasor is a complex-valued number of the form $\alpha e^{\iota\varphi}$ where α and φ are real-valued and α is non-negative.

$$\beta e^{\iota\Psi} = \boldsymbol{w}_{\mathrm{L}}^{\dagger} \boldsymbol{R} \boldsymbol{w}_{\mathrm{R}} \tag{3-8}$$

where, $\boldsymbol{w}_{\mathrm{L}}$ and $\boldsymbol{w}_{\mathrm{R}}$ are K-element vectors with the element weights in the first M slots of $\boldsymbol{w}_{\mathrm{L}}$ and the last M slots of $\boldsymbol{w}_{\mathrm{R}}$ with the non-weight slots being filled by zeros. Eqn (3-8) could equivalently be expressed as the (weighted) sum of selected elements of the covariance matrix,

$$\beta e^{\iota\Psi} = \sum_{i=1}^{K} \sum_{j=1}^{K} \boldsymbol{W} \otimes \boldsymbol{R} \tag{3-9}$$

where $\boldsymbol{A} \otimes \boldsymbol{B}$ denotes Schur (element-wise) multiplication and the weighting mask W is

$$\boldsymbol{W} = \begin{bmatrix} 0 & 0 & w_1^2 & w_1 w_2 & w_1 w_3 & w_1 w_4 \\ 0 & 0 & w_2 w_1 & w_2^2 & w_2 w_3 & w_2 w_4 \\ 0 & 0 & w_3 w_1 & w_3 w_2 & w_3^2 & w_3 w_4 \\ 0 & 0 & w_4 w_1 & w_4 w_2 & w_4 w_3 & w_4^2 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}. \tag{3-10}$$

This notation permits viewing the split-aperture approach as being a phasor derived from the covariance matrix—as is illustrated in Figure 3-2. From this perspective, we see that the split-aperture approach involves elements of the covariance matrix which do not contain any directional information (elements along the main diagonal) as well as elements which contain counterproductive information (elements below the main diagonal which cancel contributions

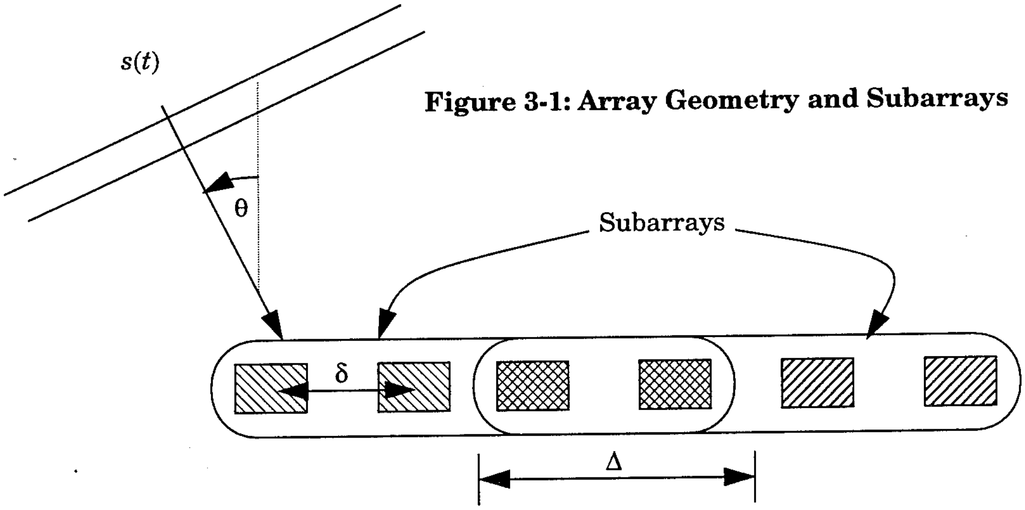


Figure 3-1: Array Geometry and Subarrays

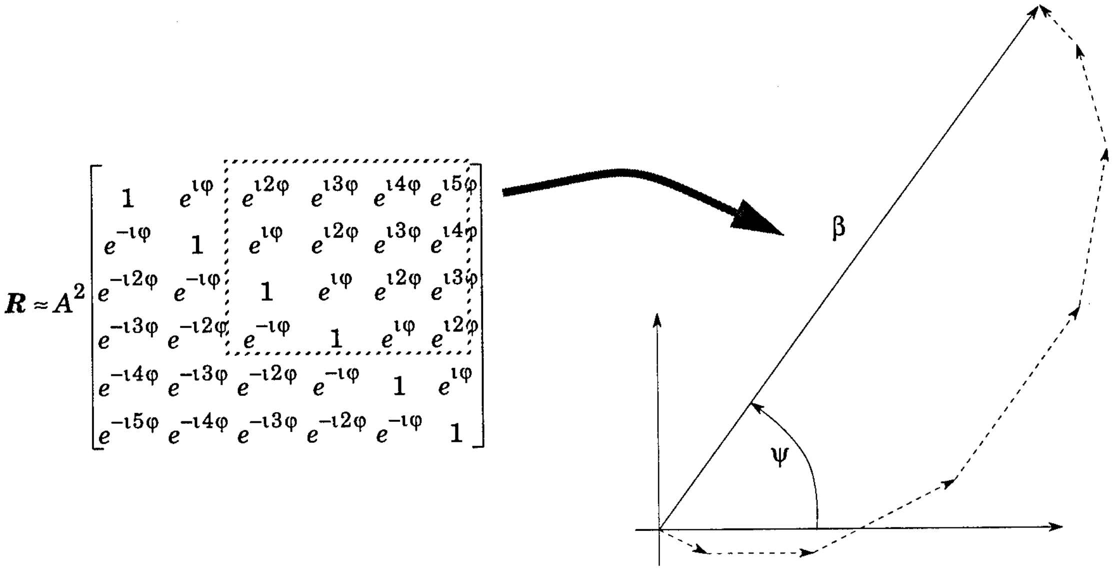
from superdiagonal elements). *However*, the split-aperture approach is *efficient* and *effective* since the composite phase Ψ may be directly mapped into the corresponding geometric angle, θ, via

$$\theta = \operatorname{asin}\left(\frac{c}{2\pi f\Delta}\Psi\right) \tag{3-11}$$

where Δ is the distance between the phase centers of the subarrays, f is the signal frequency, and c is the signal propagation speed. Although fast and easily implemented, the split-aperture approach has a phase ambiguity if the phase centers of the subarrays are offset by more than a half-wavelength; this is illustrated in Figure 3-3 for a one wavelength offset in phase centers. Although increasing the phase center offset (i.e., sensor baseline) increases the DOA estimation accuracy for sources near the beam boresight, it can also result in erroneous *and undetectable* errors in the determination of the source direction.

3.2.3 Full-Aperture DOA Estimation

Rubano [36] proposed an alternative to the split-aperture approach which avoids the phase ambiguity. This is accomplished by summing the upper triangle of the covariance matrix as illustrated in Figure 3-4. The resulting map from the phasor angle to geometric angle is illustrated in Figure 3-5 for a six-element array having half-wavelength element spacing and compared to the corresponding split-aperture mapping using one-wavelength offset phase



$$\boldsymbol{R} \approx A^2 \begin{bmatrix} 1 & e^{\iota\varphi} & e^{\iota 2\varphi} & e^{\iota 3\varphi} & e^{\iota 4\varphi} & e^{\iota 5\varphi} \\ e^{-\iota\varphi} & 1 & e^{\iota\varphi} & e^{\iota 2\varphi} & e^{\iota 3\varphi} & e^{\iota 4\varphi} \\ e^{-\iota 2\varphi} & e^{-\iota\varphi} & 1 & e^{\iota\varphi} & e^{\iota 2\varphi} & e^{\iota 3\varphi} \\ e^{-\iota 3\varphi} & e^{-\iota 2\varphi} & e^{-\iota\varphi} & 1 & e^{\iota\varphi} & e^{\iota 2\varphi} \\ e^{-\iota 4\varphi} & e^{-\iota 3\varphi} & e^{-\iota 2\varphi} & e^{-\iota\varphi} & 1 & e^{\iota\varphi} \\ e^{-\iota 5\varphi} & e^{-\iota 4\varphi} & e^{-\iota 3\varphi} & e^{-\iota 2\varphi} & e^{-\iota\varphi} & 1 \end{bmatrix}$$

Figure 3-2: Split-Aperture Phasor viewed as the Sum of Covariance Matrix Elements

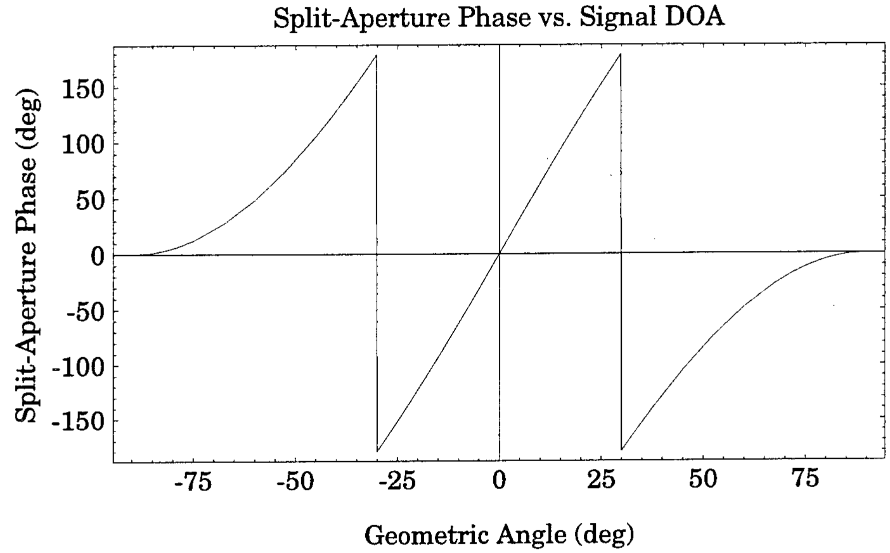


Figure 3-3: Split-Aperture DOA Mapping Ambiguity

$$\boldsymbol{R} \approx A^2 \begin{bmatrix} 1 & e^{\iota\varphi} & e^{\iota 2\varphi} & e^{\iota 3\varphi} & e^{\iota 4\varphi} & e^{\iota 5\varphi} \\ e^{-\iota\varphi} & 1 & e^{\iota\varphi} & e^{\iota 2\varphi} & e^{\iota 3\varphi} & e^{\iota 4\varphi} \\ e^{-\iota 2\varphi} & e^{-\iota\varphi} & 1 & e^{\iota\varphi} & e^{\iota 2\varphi} & e^{\iota 3\varphi} \\ e^{-\iota 3\varphi} & e^{-\iota 2\varphi} & e^{-\iota\varphi} & 1 & e^{\iota\varphi} & e^{\iota 2\varphi} \\ e^{-\iota 4\varphi} & e^{-\iota 3\varphi} & e^{-\iota 2\varphi} & e^{-\iota\varphi} & 1 & e^{\iota\varphi} \\ e^{-\iota 5\varphi} & e^{-\iota 4\varphi} & e^{-\iota 3\varphi} & e^{-\iota 2\varphi} & e^{-\iota\varphi} & 1 \end{bmatrix}$$

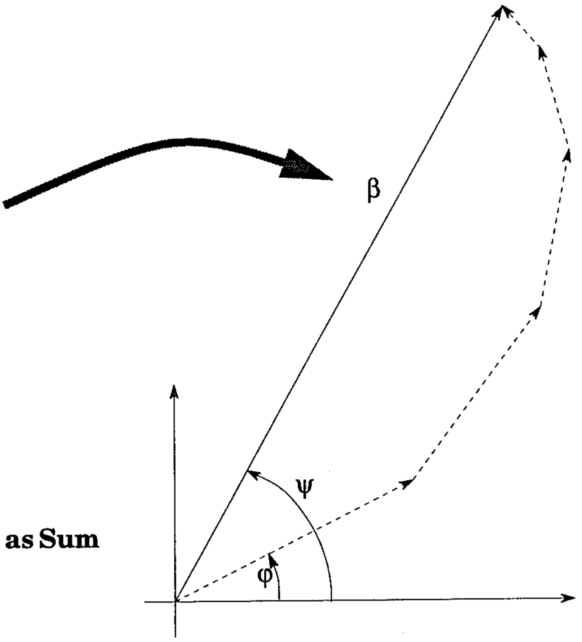


Figure 3-4: Full-Aperture Phasor viewed as Sum of Covariance Matrix Elements

centers. If we permit the use of an arbitrary weighting matrix, $\boldsymbol{W}$, then the Rubano vector, $\beta e^{\iota\Psi}$, calculated via,

$$\beta e^{\iota\Psi} = \sum_{i=1}^{K} \sum_{j=1}^{K} \boldsymbol{W} \otimes \boldsymbol{R} \tag{3-12}$$

may be seen to be quite general. In fact, the split-aperture algorithm is a special form of the full-aperture approach wherein the element weighting facilitates an easily calculated mapping from the Rubano phase, Ψ, to the associated geometric angle. In general, the mapping function does not have a closed-form representation; however, an implementation involving a lookup table is relatively straight-forward.

Although the maximal use of the covariance matrix information leads to the full-aperture approach having slightly better accuracy than the classical split-aperture approach, the advantage is not overwhelming. The real advantage of the full-aperture approach is the unambiguous mapping which avoids incorrect interpretations of the received data. Such situations might arise in multi-path scenarios involving surface or other signal reflections.

Additionally, it should be noted that—as for the split-aperture case—for maximum accuracy the covariance matrix should be electrically steered towards the source direction. Because this

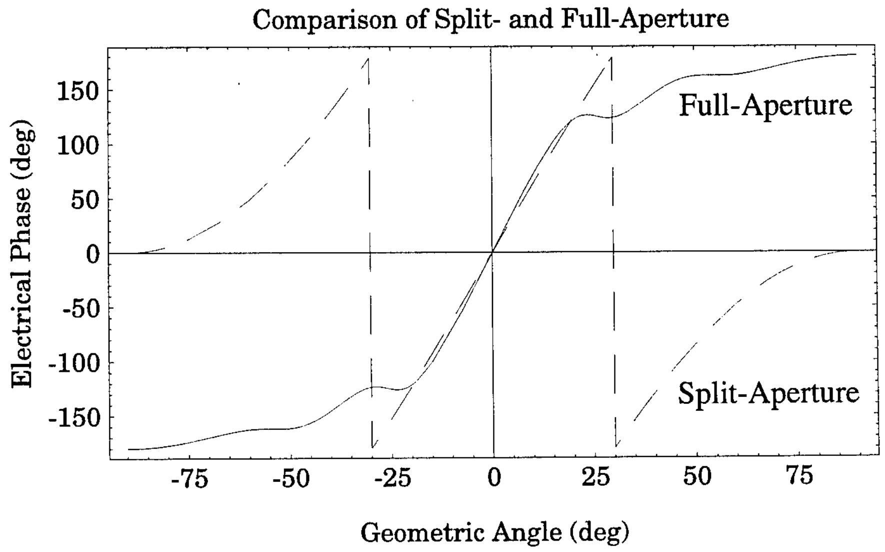


Figure 3-5: Comparison of Full-Aperture and Split-Aperture Phase Maps

steering can be easily achieved by modifying the sampled covariance matrix, the loss of sensitivity away from boresight can be mitigated.

Finally, although full-aperture and split-aperture processing assume the presence of a *single* source and, therefore, are not suitable for multi-source scenarios, the weight matrix can be chosen to provide more directivity in a hypothesized direction—at the expense of introducing inverse mapping ambiguities.

3.2.4 Direct-Mapping Signal Detection

Since the possibility exists that *no* signals are present in the sampled covariance matrix, the signal processing must ascertain whether a signal is present as well as determining a DOA. To support this, a detection criteria is required. From the physical model, there are three possible approaches to the detection problem which utilize the structure of the sampled covariance matrix:

- received power
- covariance power
- covariance directivity

The received power may be determined from the trace of the sampled covariance matrix since the main diagonal corresponds to the power received by the individual array elements. If a baseline (reference) power can be established either by *a priori* knowledge of the background noise level or via temporal analysis, signals may be detected by a comparison to this threshold—i.e., classical statistical inference. This approach requires either the array be calibrated or a "learning set" be available to determine the no-signal power levels.

If we assume that the sensor noise is spatially uncorrelated, the non-diagonal elements will converge to zero while the main diagonal elements will reflect the received signal power. Hence, the ratio of the sum of the *magnitudes* of the non-diagonal elements relative to the diagonal elements is an indicator of the presence of an impinging signal.

Alternately, we could steer the covariance matrix (i.e., electrically steer the array) towards a direction and examine the *directivity* of the covariance matrix since the covariance matrix elements resulting from a source at the array normal would be (nearly) real-valued. This detection criterion is closely aligned with the full-aperture method of DOA estimation.

In the event of a pulsed signal—i.e., active sonar or radar—the non-signal baseline level may be determined from a range-Doppler map analysis and this result used to direct the target angle determination; however, in the passive signal case, the selection of appropriate detection criteria involves *ad hoc* detectors derived from analysis of test data.

3.3 Multi-Source DOA Estimation Methods

If multiple signals are present, the split-aperture and full-aperture approaches return a *single* DOA estimate which is a composite of the sources. To determine the DOAs of multiple sources, the classical methods of beamforming, maximum entropy, or minimum variance estimation may be used. In the subsequent sections, we explore each of these as well as their relative performance against a simple, low-noise multi-source scenario. The assumed array topology in this scenario has 6-elements at half-wavelength phase-center separations. Increasing the number of elements will improve the performance of each method; however, the relative performance will remain consistent.

To reiterate earlier statements, these classical methods may be *optimal* from an engineering perspective despite being *suboptimal* from a mathematical perspective due to the reduced computational load relative to the subspace methods. Because the subspace methods exploit additional information about the array topology and signal environment, they will generally provide better accuracies; however, these classical methods may provide results which are "good enough".

3.3.1 Beamforming

Given our knowledge of the array topology, the beamforming method seeks the geometric angle(s), θ which maximizes the received signal power,

$$P_{\mathrm{BF}}(\theta) = \boldsymbol{w}(\theta)^{\dagger} \boldsymbol{R}\boldsymbol{w}(\theta) \tag{3-13}$$

where $\boldsymbol{w}$ is defined as the Schur product (element-by-element product) of a weight vector, $\boldsymbol{a}$, and the steering vector, $\boldsymbol{d}(\theta)$, defined in Eqn (3-3); thus,

$$\boldsymbol{w}(\theta) = \boldsymbol{a} \otimes \boldsymbol{d}(\theta). \tag{3-14}$$

The weighting vector is chosen to reduce the contributions which are known *a priori* to have larger sensor errors (e.g., the outer elements of a sonar array which are subject to increased flow noise) and may be complex-valued to compensate for known phase or amplitude errors in the array elements. The beamforming response behavior, $P_{BF}(\theta)$, is illustrated in Figure 3-6 for a six-element array. For a single signal source (and ideal elements and noise), beamforming is an unbiased estimator of the source direction; however, as is illustrated in Figure 3-6, this is not true if multiple sources are present. Furthermore, due to the "low resolution" of the beamforming response function, it may be difficult to distinguish/detect multiple sources using beamforming if those sources have similar DOAs or have greatly differing signal levels.

3.3.2 Maximum Entropy (Burg's Method)

Burg [56] was the first to improve on the beamforming approach; he realized that the resolution was limited by the finite number of elements—analogous to a DFT resolution being determined by the number of samples. To improve the resolution, he proposed extrapolating the covariance function beyond the nominal array size limitations. Although there are, in principle, an infinite number of possible extrapolations, Burg contended that the extrapolation should be selected which maximized the signal entropy. In formalizing this concept, Burg showed that the maximum entropy is achieved by fitting an auto-regressive (AR) model to the data. Thus, we seek the AR coefficients, a_k, which minimize the expected prediction error,

$$\min \arg E\left(\left|x_i(t) - \sum_{k=1}^{r} a_k x_{i-k}(t)\right|^2\right) \tag{3-15}$$

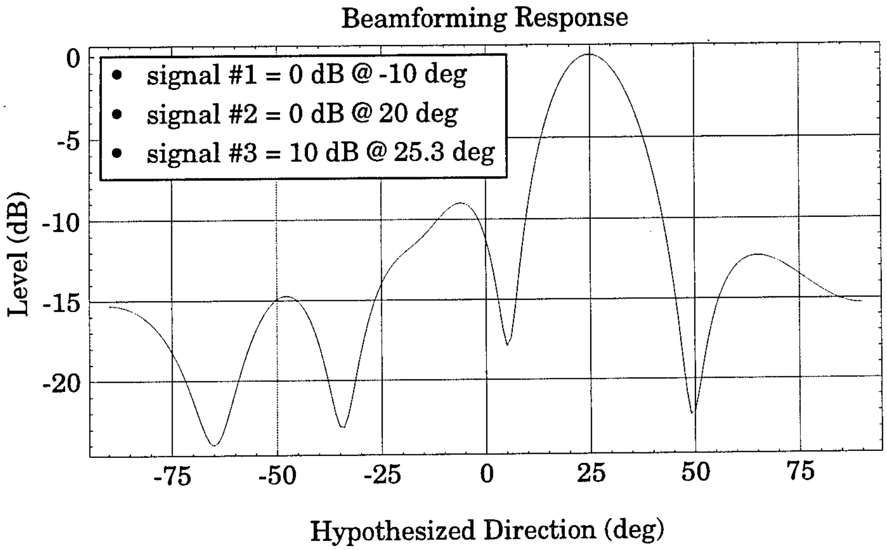


Figure 3-6: Beamforming Response Behavior for a Multi-Source Scenario

where r is the order of the AR model. Setting $r = K$, where K is the size of the sensor array, Eqn (3-15) may be represented as,

$$\boldsymbol{a} = \arg\min\ (\boldsymbol{a}^{\dagger}\boldsymbol{R}\boldsymbol{a}) \qquad (3\text{-}16)$$

subject to the constraint that the first element of the AR coefficients be “1”, i.e.,

$$\boldsymbol{a}^{\dagger}\boldsymbol{e}_1 = 1 \qquad (3\text{-}17)$$

where

$$\boldsymbol{a} = \begin{bmatrix} a_1 \ \ldots \ a_K \end{bmatrix}^{\mathrm{T}} \qquad (3\text{-}18)$$

and

$$\boldsymbol{e}_1 = \begin{bmatrix} 1 \ 0 \ \ldots \ 0 \end{bmatrix}^{\mathrm{T}}. \qquad (3\text{-}19)$$

The AR coefficients resulting from this minimization problem may be found via Lagrange multiplier techniques to be

$$\boldsymbol{a} = \frac{\boldsymbol{R}^{-1}\boldsymbol{e}_1}{\boldsymbol{e}_1^{\mathrm{T}}\boldsymbol{R}^{-1}\boldsymbol{e}_1}. \qquad (3\text{-}20)$$

The DOAs then can be found from the peaks of the spatial spectrum given by

$$P_{\mathrm{ME}}(\theta) = \frac{1}{\left|\boldsymbol{d}(\theta)^{\dagger}\boldsymbol{a}\right|^{2}}. \qquad (3\text{-}21)$$

The spatial spectrum is illustrated in Figure 3-7 for the same situation as for beamforming in Figure 3-6. Note that even though the two adjacent sources are not distinguishable, the source at -10 degrees has been localized more accurately. Furthermore, the signal sources are more easily distinguished from the non-signal peaks in the spatial response. However, this increased performance relative to beamforming is achieved at a cost of considerable increase in computational complexity and loading—especially as the size of the sensor array increases.

3.3.3 Minimum Variance (Capon's Method)

The beamforming approach has relatively poor resolution because the delay-and-sum implicit in the beamforming for a signal DOA also contains contributions from the other signals. Capon's [37] approach was to modify the delay-and-sum at a given DOA angle to minimize the contribution from interfering sources. Therefore, he proposed to estimate the j^{th} signal waveform, $s_j(t)$, by a linear estimator using the samples at the array elements, $\boldsymbol{x}(t)$, i.e.,

$$\hat{s}_j(t) = \boldsymbol{w}_j^{\dagger}\,\boldsymbol{x}(t). \tag{3-22}$$

where $\boldsymbol{w}_j$ is a K-element complex-valued vector. Since the array element signals are related to the J impinging wavefronts by

$$\boldsymbol{x}(t) = \sum_{j=1}^{J} \boldsymbol{d}_j \boldsymbol{s}_j(t) + \boldsymbol{n}(t) \tag{3-23}$$

the estimated signal may be expressed as a scaled version of the desired signal, $s_j(t)$ which is corrupted by contributions from the other signals and the noise,

$$\hat{s}_j(t) = \boldsymbol{w}_j^{\dagger}\,\boldsymbol{d}_j s_j(t) + \sum_{i \neq j} \boldsymbol{w}_j^{\dagger}\,\boldsymbol{d}_i s_i(t) + \boldsymbol{w}_j^{\dagger}\,\boldsymbol{n}(t). \tag{3-24}$$

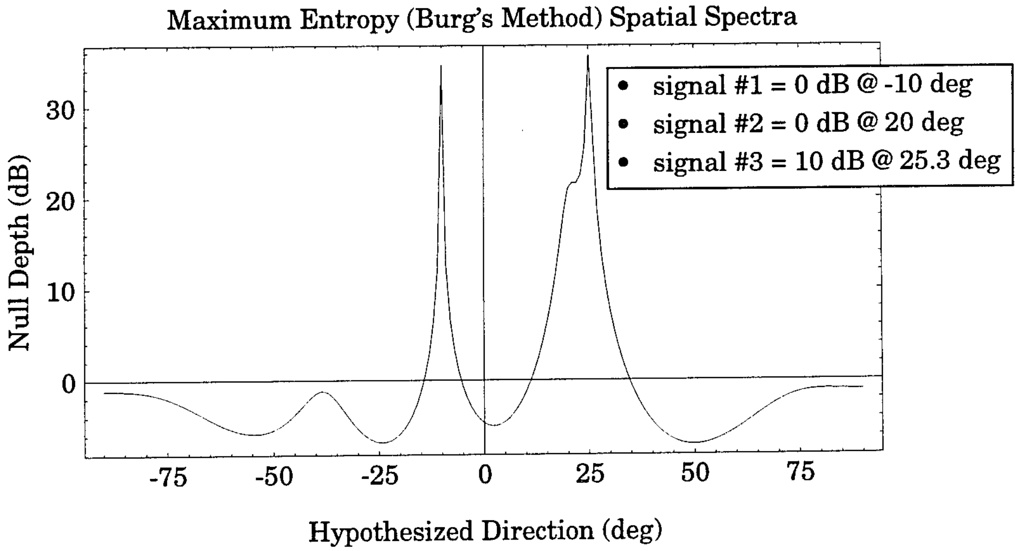


Figure 3-7: Maximum Entropy Method Response for a Multi-Source Scenario

Capon contended that minimizing the power of the estimated waveform subject to the constraint that the desired waveform be undistorted (in the sense that the array gain in the desired signal's direction be equal to unity) should improve on the beamforming resolution. Thus, we seek the solution to the minimization problem,

$$\boldsymbol{w}_j = \arg\min \boldsymbol{w}_j^\dagger \boldsymbol{R}\boldsymbol{w}_j, \tag{3-25}$$

subject to the constraint

$$\boldsymbol{w}_j^\dagger \boldsymbol{d}(\theta_j) = 1. \tag{3-26}$$

The solution to this minimization problem may be found via Lagrange multiplier techniques to be

$$\hat{\boldsymbol{w}}_j = \frac{\boldsymbol{R}^{-1}\boldsymbol{d}(\theta_j)}{\boldsymbol{d}(\theta_j)^\dagger \boldsymbol{R}^{-1}\boldsymbol{d}(\theta_j)} = \alpha^2\boldsymbol{R}^{-1}\boldsymbol{d}(\theta_j) \tag{3-27}$$

where α^2 is defined as

$$\alpha^2 = \frac{1}{\boldsymbol{d}(\theta_j)^\dagger \boldsymbol{R}^{-1}\boldsymbol{d}(\theta_j)} = \hat{\boldsymbol{w}}^\dagger \boldsymbol{R}\hat{\boldsymbol{w}} \tag{3-28}$$

and is the residual power at the processor output. Since this should be maximized if a source is truly arriving from a direction θ, Capon contended that the DOAs should correspond to peaks of Eqn (3-28). This angular power spectrum behavior is illustrated in Figure 3-8.

Capon's approach has a problem with correlated signals—due to situations like multipath propagation—since, even though the selection of the weight vectors guarantees the $\boldsymbol{w}_j^\dagger \boldsymbol{d}_j s_j(t)$ term in Eqn (3-24) will have a unity coefficient, the auxiliary terms which are minimized may effectively cancel the signal.

Borgiotti and Kaplan [41] proposed a variant of the minimum variance estimator which used a weight vector of the form

$$\boldsymbol{w}_j = \mu\boldsymbol{R}^{-1}\boldsymbol{d}(\theta_j) \tag{3-29}$$

subject to the constraint,

$$w_j^\dagger w_j = 1. \tag{3-30}$$

This modification means that the contributions of the noise to the delay-and-sum processing is (on average) the same for every direction. The resulting power output when steered to a direction θ is

$$P_{\mathrm{BK}}(\theta) = \frac{d(\theta)^\dagger R^{-1} d(\theta)}{d(\theta)^\dagger R^{-2} d(\theta)}. \tag{3-31}$$

The DOAs correspond to peaks in this power spectra—as is illustrated in Figure 3-9. For this particular scenario, the three signals are clearly distinguishable—although, not as well as the subspace methods when operating with an accurate estimate of the number of impinging signals.

3.4 Summary

Figure 3-10 shows the normalized spatial spectral response of the methods discussed in this chapter relative to that of the MUSIC algorithm presented in Chapter 5. By exploiting the subspace decomposition and the associated estimate of the number of signals, the MUSIC algorithm is better able to resolve the adjacent signals; however, it should be noted that classical methods discussed in this chapter attempt to estimate the signal power whereas the null-depth of the MUSIC algorithm corresponds to subspace stability—which is an indicator of

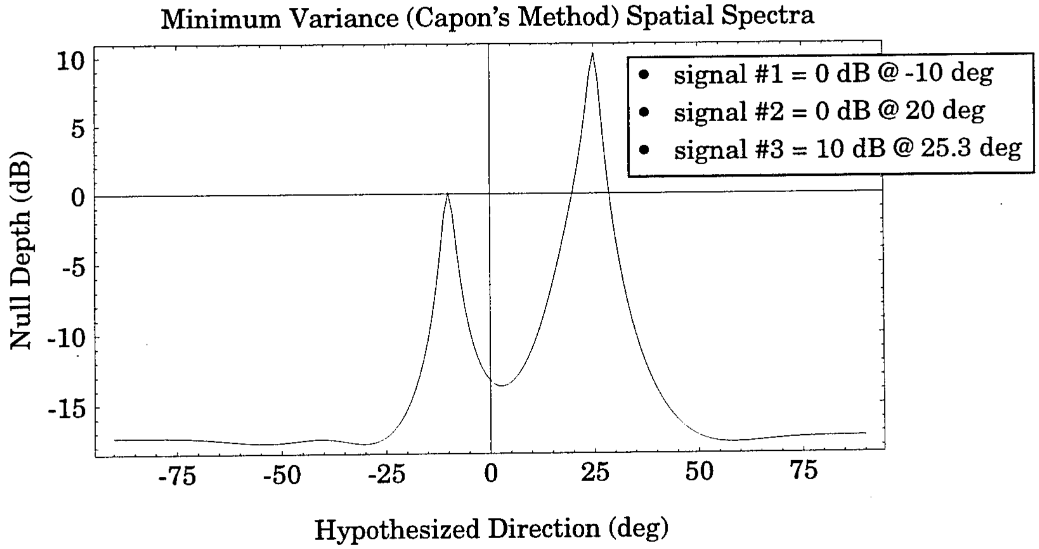


Figure 3-8: Minimum Variance Angular Power Spectra for a Multi-Source Scenario

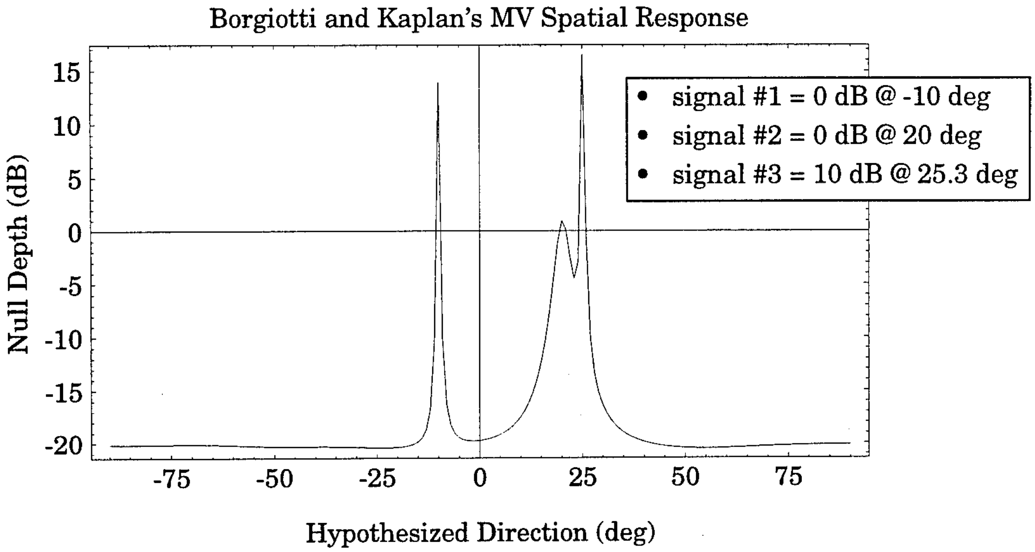


Figure 3-9: Borgiotti & Kaplan's Minimum Variance Spatial Spectra Illustration

signal strength but does not directly correspond to the levels generated using Burg's, Capon's, or Borgiotti & Kaplan's methods.

The greater resolution of the subspace methods is predicated upon an accurate estimate of the number of impinging signals whereas the classical methods do not require an *a priori* detection. Thus, at the price of decreased performance, the classical methods offer computational simplicity and efficiency; hence, given a limited number of strong equi-powered signals, the classical methods may be preferable. However, in general, the increased performance of the subspace methods offset the additional computational load. This increased performance is illustrated by Figure 5-1 (page 60) which shows the spatial response due to wider dynamic range signals.

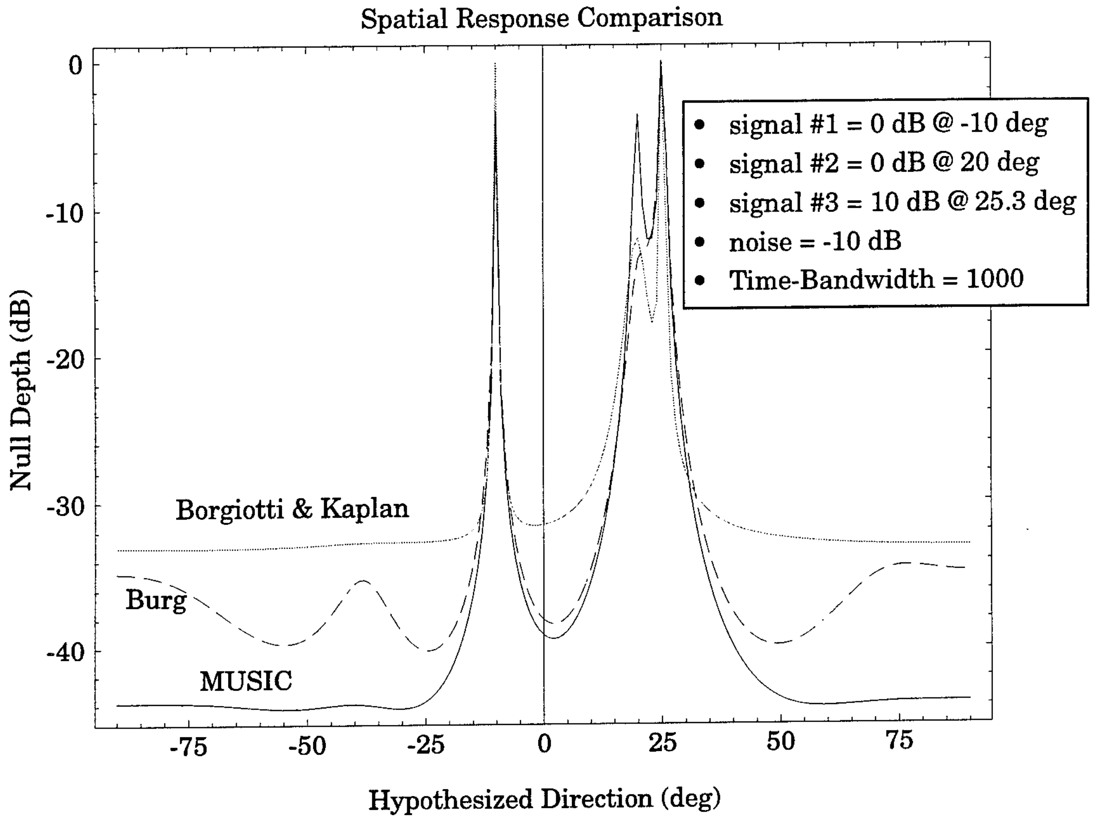


Figure 3-10: Comparison Plot of MUSIC and Classical Multi-Source Methods

4. Detection and Enumeration

Since the subspace-based methods of DOA estimation require identifying a *signal subspace* and a *noise subspace*, a central problem to these methods is the determination of the number of impinging signals from the covariance matrix—i.e., the number of dimensions associated with each subspace. Although the signal detection and DOA estimation processing should be combined for *optimal* performance (see, for example, Wax [14]), such an approach is computationally prohibitive; hence, the detection and DOA processing are generally performed sequentially rather than simultaneously. The intent of this chapter is to address the problems associated with the signal detection and outline some of the candidate algorithms to accomplish the detection.

The discussion in this chapter presumes that equi-powered band-limited Gaussian noise corrupts the received signal at each sensor in the array. Furthermore, we assume that this noise is spatially uncorrelated—i.e., the noise is neither spatially or temporally colored. (Signal detection in colored noise fields is discussed in Chapter 8.) Although numerous detection schemes have been proposed (e.g., [14]–[28]) which exploit this presumption, in many applications the assumption of uncolored noise fields is often dangerously erroneous—as demonstrated by LeCadre's data [51] from real towed sonar arrays as well as the in-water data sets used herein. As a result, although the enumeration criteria discussed in this chapter are important from a pedagogical perspective and valid for some scenarios, their practical application in sonar signal processing is suspect.

The subspace stability methods of Chapter 7 adopt a different paradigm for the signal enumeration—using subspace stability coupled with *a priori* knowledge of the array manifold rather than the energy-based methods of this chapter.

4.1 Introduction and Background

Without any noise or other error sources, the sampled covariance matrix with J uncorrelated cisoid signals impinging on a K-element array would have J non-zero eigenvalues and $K-J$ zero eigenvalues—which would make determining the number of incoming signals fairly easy. Unfortunately, the sampled covariance matrix *will* be corrupted so that distinguishing between the eigenvalues associated with the signal and those associated with the noise will be non-trivial.

The general premise of the signal detection processing is that the sampled covariance matrix may be represented as in Eqn (2-25) by,

$$\boldsymbol{R} = \boldsymbol{D}\boldsymbol{R}_s\boldsymbol{D}^\dagger + \Sigma_n = \boldsymbol{D}\boldsymbol{R}_s\boldsymbol{D}^\dagger + \sigma^2\boldsymbol{I}, \tag{4-1}$$

where $\boldsymbol{R}_s$ has a rank of J and the noise covariance matrix, Σ_n, is assumed to be a scaled identity matrix—which is equivalent to assuming that the noise at each sensor is spatially and temporally independent with equal statistics (power). Some noise structure must be assumed to permit signal detection; as discussed in Chapter 8, if the noise structure can be accurately modelled, the sampled covariance may be whitened to permit applications of these techniques.

Under this assumption, the K eigenvalues of $\boldsymbol{R}$ are

$$\lambda_k = \begin{cases} \mu_k + \sigma^2 & k = 1, 2, 3, \ldots, J \\ \sigma^2 & k = J+1, \ldots, K \end{cases}. \tag{4-2}$$

The signal eigenvalues, μ_j, represent a composite effect of the impinging signals so we cannot make simple inferences about any single signal based upon these eigenvalues (except in the single signal case).

Since the sampled covariance matrix only approximates the "true" covariance, i.e., $\hat{\boldsymbol{R}} \neq \boldsymbol{R}$, the sampled eigenvalues will be perturbed from the nominal values of Eqn (4-2) due to the effects of noise and finite sampling (averaging) intervals; therefore, with a probability of one, *none* of the K eigenvalues will be equal. For our purposes, let us order the eigenvalues from largest to smallest so that

$$\lambda_1 > \lambda_2 > \ldots > \lambda_J > \lambda_{J+1} > \ldots > \lambda_K. \tag{4-3}$$

Presumably, the $K-J$ smallest eigenvalues will be *almost equal* and the J eigenvalues associated with the signals will be "significantly" larger than the noise eigenvalues—the only problem is to define an algorithm which understands the distinction.

The initial approach to the recognition problem was a sequence of hypothesis tests based upon the work of Bartlett and Lawley in the 1950s. The problem with this approach is the selection of a threshold is somewhat arbitrary and requires a subjective decision by the designer of the test. To avoid this subjective threshold setting, *information theoretic criteria* were developed by Akaike, Rissanen, Schwartz, Hannan, and others; these are known as the *Akaike Information Criteria* (AIC) and the *Minimum Description Length* (MDL) criteria. Wax [14] and Kailath [15] derived the eigenvalue forms of these criteria. The AIC is not a *consistent estimator* so even at high SNRs it will tend to over-estimate the number of signals while the MDL tends to under-estimate the number of signals at low SNRs. Recently, Chen, Wong, and Reilly [24] have proposed a consistent estimator (the Eigen-Threshold Approach) which attempts to avoid the deficiencies of the MDL via a controllable detection parameters. These methods are discussed in more detail in the following sections.

4.1.1 Detection Assumptions

The Bartlett-Lawley, AIC, MDL, and MIC approaches make assumptions which are often invalid in real-world applications. These assumptions include:

- Spatially uncorrelated noise;
- Equal noise power at each sensor;
- Uncorrelated signals.

If correlated signals are suspected or likely, spatial smoothing (Chapter 9) may be employed to remove that concern from the detection processing. However, the assumption of uncorrelated sensor noise with equal power is a fundamental modelling assumption exploited by these algorithms. LeCadre [51], Wax [52], Fuchs [53], and Zhang and Wong [54] have done some work on detection in spatially correlated noise.

4.1.2 Terms and Definitions

The detection theory literature is sprinkled with terms which are not very intuitive—at least to this author. The intent of this section is to collect some of the notation and terms which are used in the referenced papers and books. These are presented in Table 4-1.

4.2 The Maximum Likelihood Statistic

The likelihood statistic is very important in the estimation of the number of impinging signals. As derived by Wax [14], if we assume that the *complex-valued* samples are independent, zero-mean Gaussian, then the probability density associated with a sequence of N samples at our K-element array, $\boldsymbol{x}(t_i)$, is the multi-variate Gaussian,

$$f(\boldsymbol{X}|\,\theta^{(j)}) = f(\boldsymbol{x}(t_1), \ldots, \boldsymbol{x}(t_N)|\,\theta^{(j)}) = \prod_{i=1}^{N} \frac{1}{\pi^K |\boldsymbol{R}^{(j)}|} e^{-\boldsymbol{x}(t_i)^\dagger [\boldsymbol{R}^{(j)}]^{-1} \boldsymbol{x}(t_i)} \tag{4-4}$$

Table 4-1: Detection Theory Terms and Definitions

Term	*Notation*	*Definition*
null hypothesis	H_0	the hypothesis that one of the two possible (in a binary decision) states is in effect. Although the definition of this *null hypothesis* is arbitrary, it is customary to define it as the absence of the condition or signal of interest.
event hypothesis	H_1	the complement to the null hypothesis—i.e., the hypothesis that the signal or condition of interest *is* in effect.
observations	$\boldsymbol{X}$	the set of N observations made at each of the K sensors in the array.
model parameters	θ_i	the model parameter states associated with the H_i hypothesis.
size	α or P_{FA}	the *probability of false alarm* in testing a binary hypothesis—also known as a *type I error.*
power	β or P_{D}	the *probability of detection* in a binary decision.
miss	$P_{\text{M}} = 1 - P_{\text{D}}$	the probability of *missing* a detection in a binary decision.
density function	$f_{\theta_i}(\boldsymbol{X}) = f(\boldsymbol{X}\vert\theta_i)$	probability density function conditioned on the hypothesis H_i—in other words, the density function presuming the hypothesis is correct.
maximum likelihood estimate	$\hat{\theta} = \max(f_{\theta}(\boldsymbol{X}))$	the maximum likelihood (most probable) estimate of the model parameters given the sampled data set, $\hat{\boldsymbol{x}}$.
log likelihood statistic	$L_j = \ln(f_{\hat{\theta}^{(j)}}(\boldsymbol{X}))$	*Likelihood* of hypothesis H_j given the sampled data.
consistency		the characteristic that a detector is asymptotically correct.

where $\theta^{(j)}$ denotes the parameter vector describing the j impinging signals' eigenvalues and eigenvectors as well as the noise power[1],

[1] We have adopted the notation that "j" represents a hypothesized number of signals as opposed to "J" which is the true number of impinging wavefronts.

$$\theta^{(j)} = \left[\lambda_1 \ldots \lambda_j \, \sigma^2 \, \boldsymbol{b}_1^{\mathrm{T}} \ldots \boldsymbol{b}_j^{\mathrm{T}}\right]^{\mathrm{T}} \qquad (4\text{-}5)$$

and $\boldsymbol{R}^{(j)}$ is the *true* covariance matrix (vice the *sampled* covariance matrix, $\hat{\boldsymbol{R}}$). Wax noted that of the $j+1+2jK$ parameters in $\theta^{(j)}$, we only have $j\,(2K-j)+1$ degrees of freedom due to the constraints that the eigenvectors have a unit norm and be mutually orthogonal. The number of degrees of freedom will be important in the following detection criteria. Using matrix notation, Eqn (4-4) may be equivalently presented as,

$$f(\boldsymbol{X}|\,\theta^{(j)}) = \frac{1}{\pi^{KN}|\boldsymbol{R}|^{N}} e^{-N\mathrm{tr}(\boldsymbol{R}^{-1}\hat{\boldsymbol{R}})}. \qquad (4\text{-}6)$$

The maximum likelihood estimator of $\theta^{(j)}$ is achieved by maximizing the likelihood function with respect to $\theta^{(j)}$; i.e., choosing the parameter set, θ, such that the actual observation distribution matches the predicted distribution based on *a priori* assumptions about the probability density functions. Following the derivation of Wax [14] leads us to conclude that

$$\hat{\boldsymbol{b}}_k = \boldsymbol{u}_k \qquad k = 1, 2, \ldots, j \qquad (4\text{-}7)$$

$$\hat{\lambda}_k = l_k \qquad k = 1, 2, \ldots, j \qquad (4\text{-}8)$$

and

$$\hat{\sigma}^2 = \frac{1}{K-j} \sum_{i=j+1}^{K} l_i. \qquad (4\text{-}9)$$

where l_i are the eigenvalues derived from the sampled covariance matrix, $\hat{\boldsymbol{R}}$, and $\boldsymbol{u}_i$ are the eigenvectors associated with the j largest eigenvalues. In other words, assuming a zero-mean Gaussian distribution, the best estimate of the signal eigenvalues and eigenvectors are those derived from the sampled covariance matrix and the estimate of the noise power is simply the average of the remaining $K-j$ sampled covariance matrix eigenvalues.

Substituting the maximum likelihood estimates of Eqn (4-7)–Eqn (4-9) into the probability density of Eqn (4-6) and taking the log yields the log likelihood statistic,

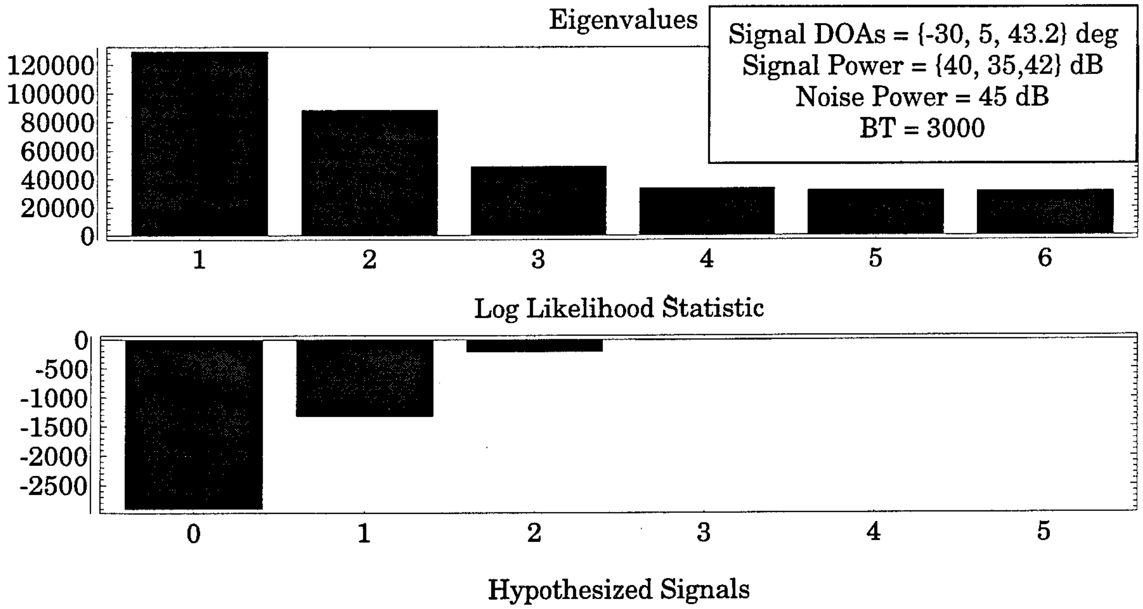


Figure 4-1: Example Eigenvalues and Associated Likelihood Ratios

$$L_j = \ln\left(\left(\frac{\prod_{i=j+1}^{K} \hat{\lambda}_i^{\frac{1}{K-j}}}{\frac{1}{K-j} \sum_{i=j+1}^{K} \hat{\lambda}_i} \right)^{(K-j)N} \right) \tag{4-10}$$

where N is the number of *independent* samples (approximately the time-bandwidth product if we assume band-limited Gaussian noise) and we have adopted the notation that $\hat{\lambda}_i = l_i$.

L_j actually tests whether the non-signal eigenvalues are unique—i.e., it is a measure of the size of the orthogonal subspace; thus, L_j is a monotonically increasing function which checks the (log of the) likelihood that the $K-j$ smallest eigenvalues are equal. This likelihood is the ratio of the geometric-mean to the arithmetic-mean of the $K-j$ smallest eigenvalues; if these eigenvalues were equal, the ratio would be identically "1". Since additive noise will ensure that each eigenvalue is unique, we must define some criteria to declare that the $K-J$ eigenvalues are "effectively equal"; this is the topic of the following sections.

Figure 4-1 illustrates the ML behavior as well as the eigenvalues associated with three signals arriving at a six-element array. Note that the log likelihood achieves "zero" by hypothesizing

$K-1$ signals—thereby concluding that *all* of the eigenvalues are unique. Also note that the probability that there are *no* signals represented in the sampled covariance matrix is fairly low.

4.3 The Bartlett-Lawley Sequential Testing Approach

The Bartlett-Lawley approach involves sequentially testing the hypothesis, H_j, that the eigenvalues represent j signals and $K-j$ equal-valued noise eigenvalues until some pre-specified threshold is exceeded. A difficulty with this approach is that the threshold selection involves an *ad hoc* decision by the designer of the test. If the samples follow a Gaussian distribution, as the number of samples increases the statistic $-2\ln(L_j)$ will asymptotically approach a Chi-squared distribution with M degrees-of-freedom, χ_M^2, where M is the "difference in the dimension of the subspaces spanned by the parameter vector under the two hypotheses" [14]. Here this dimension is $(K-j)^2-1$. Under these assumptions, for large numbers of samples, an appropriate test which has a size (probability of false alarm) $\alpha = P_{\mathrm{FA}}$ is to reject the H_j hypothesis if

$$-2\left(\ln(L_j)\right) > c\left(\alpha; \left((K-j)^2-1\right)\right) \tag{4-11}$$

where $c(\alpha;M)$ is the upper 100α percent point of the χ_M^2 distribution.

The detection process initially assumes that no signals are present and sequentially tests the hypothesis that there are j signals until the threshold is passed. Note that the selection of an appropriate false alarm probability is left to the *subjective* assessment of the test designer. This problem is obviated by the information theoretic criteria approach first introduced by Akaike—providing the assumed probability distributions are in effect.

4.4 Information Theoretic Criteria

Because the log-likelihood statistic is monotonic, the "best" conclusion would be that the sampled covariance represents $K-1$ signal sources plus noise. Therefore; we need to modify the log-likelihood statistic with a *penalty function* to yield a more reasonable signal enumeration. This penalty function should consider the number of samples, N, used to form the sampled covariance (i.e., our confidence in the sampled structure) as well as the number of hypothesized signals, j, and their relative amplitudes. Thus, in general, information theoretic criteria have the form,

$$\text{criterion} = -\ln(f(\boldsymbol{X}|\,\theta^{(j)})) + p(n, N) \tag{4-12}$$

where n is the number of free adjusted parameters—i.e., degrees of freedom—in the model, θ. Minimizing the number of free parameters essentially corresponds to choosing the simplest valid model. As discussed in Section 4.2, this is related to the number of signals, j, and array size, K by

$$n = j(2K-j)+1. \tag{4-13}$$

Eqn (4-12) employs the *negative* of the log-likelihood statistic so our problem is to find the *minimum* of the criterion. In sum, we have converted the detection problem into one of model selection given a set of N (independent) observations and a family of models—characterized by a parameterized family of probability densities, $f(\mathbf{X}|\,\theta)$. The best fitting model is determined by the log-likelihood statistic and the penalty function.

4.4.1 Consistency

Wax [14] showed that for a penalty function to be consistent, it should be of the form,

$$p(n,N) = n\alpha(N) \tag{4-14}$$

where

$$\lim_{N \to \infty} \alpha(N) \to \infty \tag{4-15}$$

and

$$\lim_{N \to \infty} \frac{\alpha(N)}{N} \to 0. \tag{4-16}$$

Zhao, *et al.* [16] subsequently showed that consistency could be obtained if the penalty function satisfied Eqn (4-16) as well as,

$$\lim_{n \to \infty} \frac{\alpha(N)}{\ln(\ln(N))} \to \infty. \tag{4-17}$$

Consistency implies that the detector is asymptotically accurate. Note that the AIC criterion discussed later is *not* consistent in that it tends to overestimate the number of signals. Conversely, for smaller sample sets, it tends to perform better than the MDL which—despite being consistent—tends to miss signals at low SNRs. As a result, much of the literature is geared towards surpassing the AIC limited-sample performance while maintaining the asymptotic consistency. Although the asymptotic behavior may not be achievable for sonar applications, it may be achievable for sensors having higher information rates.

4.4.2 AIC—Akaike Information Criterion

Akaike's information theoretic criterion was to select the model (number of signals) which minimized the AIC[2],

$$\mathrm{AIC} = -\ln(f(\boldsymbol{X}|\,\hat{\theta})) + n \qquad (4\text{-}18)$$

where $\hat{\theta}$ is the maximum likelihood estimate of the parameter vector and n is the number of free adjusted parameters in θ. Akaike viewed the penalty term, n, as a bias correction term which made "the AIC an unbiased estimate of the mean Kulback-Liebler distance between the modelled density $f(\boldsymbol{X}|\,\theta)$ and the estimated density $f(\boldsymbol{X}|\,\hat{\theta})$" [15]. Making the appropriate substitutions and dropping constant terms leads to the AIC in terms of a hypothesized number of signals, j,

$$\mathrm{AIC}(j) = -\ln\left(\left(\frac{\prod_{i=j+1}^{K} \hat{\lambda}_i^{\frac{1}{K-j}}}{\frac{1}{K-j}\sum_{i=j+1}^{K} \hat{\lambda}_i}\right)^{(K-j)N}\right) + j(2K-j) + 1. \qquad (4\text{-}19)$$

Despite not being consistent, the AIC is a reasonably effective detection criterion. This is illustrated in Figure 4-2 where the correct number of signals (3) are estimated from a simulated covariance matrix.

4.4.3 MDL—Minimum Description Length

After Akaike's pioneering work, the MDL criterion was independently developed by Schwartz and Rissanen—via different viewpoints. Schwartz approached the problem from a Bayesian perspective—arguing that if each model is assigned an *a priori* probability, the selected model should be the one with the maximum *a posteriori* probability. Alternately, Rissanen used information theoretic arguments—contending that since each model can be used to encode the observed data, the selected model should be the one that yields the minimum code length. In the *large sample limit*, the two approaches produce the same criterion,

[2] Akaike originally termed this "An Information Criterion"; however, subsequently, "AIC" has become an acronym for "Akaike's Information Criterion".

$$\mathrm{MDL} = -\ln(f(\boldsymbol{X}|\,\hat{\theta})) + n\frac{\ln(N)}{2} \qquad (4\text{-}20)$$

which differs from the AIC criterion by a factor of $\ln(N)/2$ in the penalty term. Making the appropriate substitutions leads to an expression of the MDL in terms of the hypothesized number of signals,

$$\mathrm{MDL}(j) = -\ln\left(\left(\frac{\prod_{i=j+1}^{K} \hat{\lambda}_i^{\frac{1}{K-j}}}{\frac{1}{K-j}\sum_{i=j+1}^{K} \hat{\lambda}_i}\right)^{(K-j)N}\right) + \frac{1}{2}[j(2K-j)+1]\ln(N). \qquad (4\text{-}21)$$

The behavior of the MDL criterion is illustrated in Figure 4-2. Due to the $\ln(N)$ term in the penalty function, for reasonably large sampling intervals the MDL will tend to prefer smaller estimates of the number of signals than the AIC criterion and, therefore, may not detect marginal SNR signals. Conversely, the AIC does not adjust to reflect increased confidence in the eigenvalue estimates due to increased averaging intervals and, hence, tends to overestimate the number of signals in the high SNR scenarios.

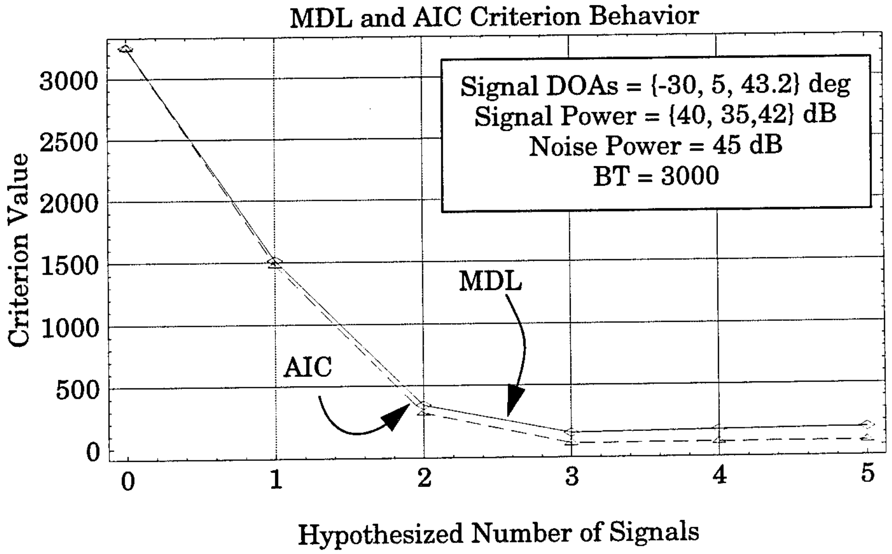


Figure 4-2: MDL and AIC Criterion Behavior—against Simulated Data .

4.4.4 MIC—Modified Information Criterion

Wong, *et al.* [23] attempted to meld the low-SNR performance of the AIC with the high-SNR performance of the MDL criterion. To do this, they realized that Wax viewed the eigenvectors of the sampled covariance matrix as an estimation parameter—which implied that the array orientation should affect the probability of detection; therefore, the covariance matrix eigenvectors should be viewed as *nuisance* parameters. In fact, this observation turns out to be correct—which reduces the parameter space of the detection problem from the $j+1+2jK$ of Wax down to the $Jj+1$ parameters associated with the signal eigenvalues as well as that of the noise,

$$\theta(j) = \left[\lambda_1\ \lambda_2\ \ldots\ \lambda_j\ \sigma^2\right]. \tag{4-22}$$

Assuming zero-mean Gaussian distributions and following some intricate manipulations, given the eigenvalues of the sampled covariance matrix, l_k, for j hypothesized signals the maximum-likelihood estimates of the signal eigenvalues and noise eigenvalue may be found by solving the set of *nonlinear simultaneous* equations,

$$\hat{\lambda}_m = l_m - \frac{\hat{\lambda}_m}{N}\sum_{\substack{i=1\\ i\neq m}}^{j}\left(\frac{\hat{\lambda}_i}{\hat{\lambda}_m - \hat{\lambda}_i}\right) - \frac{K-j}{N}\frac{\hat{\lambda}_m\hat{\sigma}^2}{\hat{\lambda}_m - \hat{\sigma}^2} \qquad m = 1, \ldots, j \tag{4-23}$$

and

$$\hat{\sigma}^2 = \frac{1}{K-j}\sum_{i=j+1}^{K} l_i + \frac{1}{N}\sum_{i=1}^{j}\frac{\hat{\lambda}_i\hat{\sigma}^2}{\hat{\lambda}_i - \hat{\sigma}^2}. \tag{4-24}$$

These equations may be solved via Newton's method [70]. For large values of N ($N > 100$), the computational loading may be reduced by adopting a gradient approach. In this case, estimates of the eigenvalues may be achieved by iterating on the difference equation,

$$\hat{\lambda}^{(i+1)} = \hat{\lambda}^{(i)} - \gamma \boldsymbol{D}(\hat{\lambda}^{(j)}) \tag{4-25}$$

where γ is a damping coefficient (typically, $1/4 < \gamma \le 1$) which controls the convergence of the iteration and $\boldsymbol{D}$ is the difference between the left and right sides of Eqn (4-23) and Eqn (4-24), i.e.,

$$D_m = \hat{\lambda}_m + \frac{\hat{\lambda}_m}{N} \sum_{\substack{i=1 \\ i \neq m}}^{j} \frac{\hat{\lambda}_i}{\hat{\lambda}_m - \hat{\lambda}_i} + \frac{K-j}{N} \frac{\hat{\lambda}_m \hat{\sigma}^2}{\hat{\lambda}_m - \hat{\sigma}^2} - l_m \qquad m = 1, \ldots, j \tag{4-26}$$

and

$$D_m = \hat{\sigma}^2 - \left(\frac{1}{K-j} \sum_{i=j+1}^{K} l_i + \frac{1}{N} \sum_{i=1}^{j} \frac{\hat{\lambda}_i \hat{\sigma}^2}{\hat{\lambda}_i - \hat{\sigma}^2} \right) \qquad m = j+1. \tag{4-27}$$

Under the new parameter set, the log likelihood function will differ from the one discussed in Section 4.2; thus, the cost function to be minimized will have the form,

$$\mathrm{MIC}(j) = -\ln(f(l_1, \ldots, l_K | \hat{\lambda}_1, \ldots, \hat{\lambda}_j, \hat{\sigma}^2)) + p(j, N). \tag{4-28}$$

Ignoring terms for the log-likelihood which do not involve the estimates of the eigenvalues or degrees of freedom leads to the equivalent criterion,

$$\mathrm{MIC}(j) = \Lambda_a + \Lambda_c + p_a + p_c + p \tag{4-29}$$

where,

$$\Lambda_a = N\left(\sum_{i=1}^{j} \frac{l_i}{\hat{\lambda}_i} + \sum_{i=j+1}^{K} \frac{l_i}{\hat{\sigma}^2} \right) + (N-K+1) \sum_{i=1}^{j} \ln(\hat{\lambda}_i) + (K-j)\,(N-j) \ln(\hat{\sigma}^2), \tag{4-30}$$

$$\Lambda_c = \sum_{\substack{i, m = 1 \\ i < m}}^{j} \ln(\hat{\lambda}_i - \hat{\lambda}_m) + \sum_{i=1}^{j} (K-j) \ln(\hat{\lambda}_i - \hat{\sigma}^2) - \sum_{\substack{i, m = j+1 \\ i < m}}^{K} \ln(l_i - l_m), \tag{4-31}$$

$$p_a = \frac{1}{2} j\,(2K - j - 1) \ln(N), \tag{4-32}$$

$$p_c = -\sum_{i = K-j+1}^{K} \ln(\Gamma(i)), \tag{4-33}$$

and,

$$p = \begin{cases} j & \text{AIC Criterion} \\ \frac{1}{2} j \ln(N) & \text{MDL Criterion} \end{cases} \tag{4-34}$$

0.2 seconds, are used to derive the sequence of covariance matrices used in the trajectory. As the array approaches the signal sources, the signal levels increase (due to propagation effects) and, therefore, the signal eigenvalues also increase. However, propagation-based wavefront scattering results in the "noise eigenvalues" being coupled to the signal eigenvalues! As a result, the non-signal eigenvalues grow with increasing signal strength rather than being independent (and approximately constant) as would be the case if the classical detection criteria were in effect. Under this situation, the information theoretic criteria conclude that *all* of the eigenvalues are unique and, therefore, *five* signals are identified rather than the correct value of *two*. However, two signals sources are clearly discernible—and, therefore, detectable given the appropriate detection criterion. The considerable jitter in the derived eigenvalue trajectory may be reduced by increasing the averaging interval by a factor of five—producing the eigenvalue trajectory of Figure 4-3. The increased averaging interval mitigates interference effects of the sampled real-valued signal.

4.6 Signal Detection—Summary and Conclusion

In this chapter, we have summarized several hypothesis testing and information theoretic approaches to the problem of recognizing impinging signals by their influence on the eigenvalues of the sampled covariance matrix. These approaches are attractive because they partition the detection and DOA estimation problem into computationally feasible segments; however, they also presume Gaussian background noise which is uncorrelated between sensors

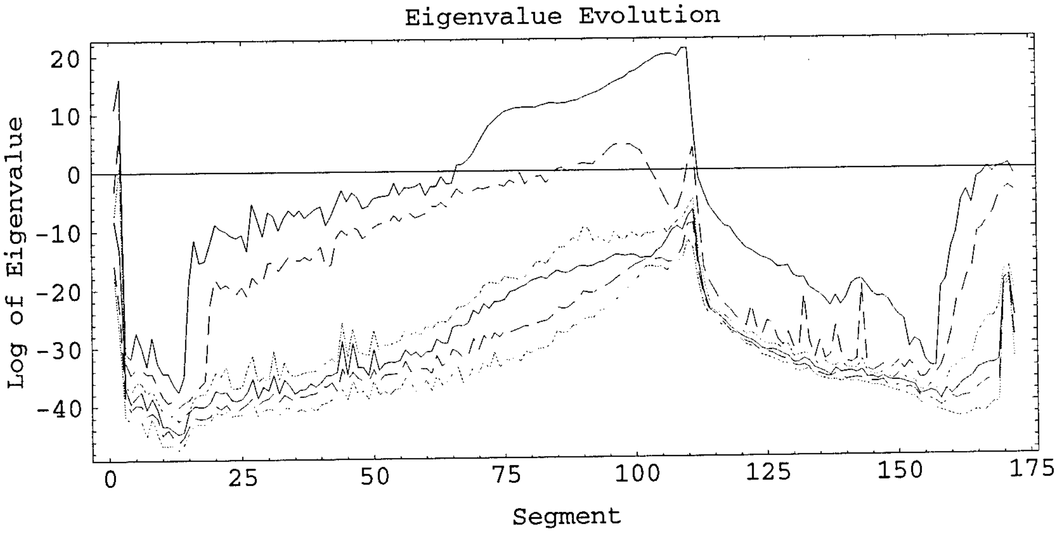


Figure 4-4: Eigenvalue Trajectories with an Increased Averaging Interval

as well as uncorrelated signals. If these assumptions are seriously violated, additional processing is required for accurate detection.

Under an alternate assumption of static source-array geometries, enumeration criteria exploiting the subspace stability may be used. Approaches based on this concept are developed in Chapter 7 wherein characteristics and performance relative to the classical information theoretic techniques are also discussed.

The DOA estimation algorithms presented in the following chapters use the estimated number of signals to partition the covariance matrix space into subspaces (represented by sets of eigenvectors) associated with the impinging signals as well as the background noise. Their performance is typically *very* sensitive to accurate estimates of the number of impinging wavefronts. As a result, the signal detection problem appears to be the limiting factor in the performance of subspace processing in operational systems.

The partitioning of the detection and estimation problem into distinct entities will be—almost by definition—suboptimal in the mathematical sense. However, from the engineering sense which considers the additional constraints of processing load, memory demands, and real-time processing requirements within the context of a "good enough" accuracy criterion, such an approach may be viewed as optimal!

5. Orthogonal Subspace Methods

The subspace-based Direction-of-Arrival (DOA) estimation algorithms may be characterized as either looking for signals orthogonal to the noise subspace (e.g., MUSIC) or operating directly on the signal subspace (ESPRIT, GEESE, etc.). In this chapter we address those techniques which exploit the orthogonality to the noise subspace. The algorithms to be explored are:

- MUSIC,
- Root-MUSIC,
- Pisarenko's Method, and
- Min-Norm.

Most of these algorithms can be implemented as an extremal-search and are, therefore, suitable for arbitrary array geometries. For uniform linear arrays root-MUSIC and Min-Norm can be implemented via polynomial rooting techniques for increased resolution and computational efficiency; however, for such array geometries, it appears that some of the signal subspace techniques (most notably, GEESE) are more robust.

Although Pisarenko's method could be viewed as one of the "classical techniques" discussed in Chapter 3, it is discussed here since it may be viewed as non-optimal implementation of the MUSIC algorithm which attempts to avoid the signal enumeration problem.

5.1 Assumptions

Under the traditional subspace processing assumptions, the sampled array covariance matrix may be modelled as

$$\boldsymbol{R} = \boldsymbol{D}\boldsymbol{R}_s\boldsymbol{D}^{\dagger} + \sigma^2\boldsymbol{I}. \tag{5-1}$$

As discussed in Chapter 2, $\boldsymbol{D}$ is a matrix which maps J wavefronts received from the directions θ at a reference location onto the elements of the array, i.e.,

$$\boldsymbol{x}(t) = \boldsymbol{D}(\theta)\boldsymbol{s}(t) + \boldsymbol{n}(t), \tag{5-2}$$

with $\boldsymbol{n}(t)$ being additive noise. $\boldsymbol{R}_s$ is the *wavefront* covariance matrix; assuming the wavefronts are uncorrelated, this corresponds to the signal covariance matrix and is of rank J.

The representation of Eqn (5-1) implies

- narrowband signals—so that the geometric and temporal contribution of the impinging wavefronts can be separated;
- spatially independent, equi-powered additive white Gaussian noise.

An eigendecomposition of $\boldsymbol{R}$ yields,

$$\boldsymbol{R} = \boldsymbol{B}\Lambda\boldsymbol{B}^{\dagger} = \sum_{k=1}^{K} \lambda_k \boldsymbol{b}_k \boldsymbol{b}_k^{\dagger}, \tag{5-3}$$

where Λ is the *ordered* diagonal eigenvalue matrix,

$$\Lambda = \begin{bmatrix} \lambda_1 & & & & \\ & \lambda_2 & & 0 & \\ & & \dots & & \\ & 0 & & \lambda_{K-1} & \\ & & & & \lambda_K \end{bmatrix} \tag{5-4}$$

$$\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_K \tag{5-5}$$

and $\boldsymbol{B}$ is the matrix of eigenvectors,

$$\boldsymbol{B} = \begin{bmatrix} \boldsymbol{b}_1 \; \boldsymbol{b}_2 \; \boldsymbol{b}_3 \; \dots \; \boldsymbol{b}_K \end{bmatrix}. \tag{5-6}$$

The noise assumption implies that the additive noise is a "ball" in the K-dimensional space—and therefore has no preferred orientation. On top of this ball, a lower-dimensional signal contribution is added which provides a preferred orientation for the eigendecomposition.[1] Thus, the eigenvalues are

$$\lambda_k = \begin{cases} \mu_k + \sigma^2 & k = 1, 2, 3, \ldots, J \\ \sigma^2 & k = J+1, \ldots, K \end{cases} . \tag{5-7}$$

Since the steering vectors, $\boldsymbol{d}(\theta)$, which map a signal wavefront onto the receive array need not be orthogonal, the magnitude of the μ_j will generally not map directly into the signal power. Eqn (5-7) implies that the eigenvectors may be partitioned into a signal subspace and an orthogonal subspace[2]

$$\boldsymbol{B} = \left[\boldsymbol{B}_s \; \boldsymbol{B}_o\right] \tag{5-8}$$

where $\boldsymbol{B}_s$ corresponds to the first J eigenvectors and $\boldsymbol{B}_o$ contains the remaining eigenvectors. To see that the subspace defined by the steering matrix, $\boldsymbol{D}$ is orthogonal to $\boldsymbol{B}_o$, recall that due to the mutually orthogonal nature of the eigenvectors,

$$\boldsymbol{R}\boldsymbol{b}_k = \lambda_k \boldsymbol{b}_k = \sigma^2 \boldsymbol{b}_k, \qquad k > J. \tag{5-9}$$

However, from the definition of $\boldsymbol{R}$ in Eqn (5-1), we see,

$$\boldsymbol{R}\boldsymbol{b}_k = (\boldsymbol{D}\boldsymbol{R}_s\boldsymbol{D}^\dagger + \sigma^2\boldsymbol{I})\,\boldsymbol{b}_k = \sigma^2 \boldsymbol{b}_k, \qquad k > J. \tag{5-10}$$

For both of these equations to be satisfied, the implication is,

$$\boldsymbol{D}\boldsymbol{R}_s\boldsymbol{D}^\dagger\,\boldsymbol{b}_k = 0 \qquad k = J+1, \ldots, K \tag{5-11}$$

and, since $\boldsymbol{D}$ and $\boldsymbol{R}$ are of full rank,

$$\boldsymbol{D}^\dagger\,\boldsymbol{b}_k = 0 \qquad J < k \le K. \tag{5-12}$$

[1] Recall that eigendecomposition is sometimes referred to as principle component analysis and for a Hermitian (i.e., positive semi-definite) matrix, the decomposition determines the principle moments of inertia and associated axes of the matrix.

[2] This terminology is derived from the fact that the eigenvectors of a Hermitian matrix are mutually orthogonal. The orthogonal subspace is sometime referred to as the "noise subspace". Since the noise, in fact, spans the entire space, such is a somewhat misleading and inaccurate terminology. The terms "orthogonal subspace" and orthogonal eigenvectors" are intended to refer to $\boldsymbol{B}_o$ and $\boldsymbol{b}_k$, $k > J$, respectively.

Equivalently, with $\boldsymbol{d}_j = \boldsymbol{d}(\theta_j)$ being the steering vector associated with the j^{th} wavefront (i.e., the j^{th} column of $\boldsymbol{D}$) which maps the wavefront received at the reference location onto the array elements,

$$\boldsymbol{d}_j^{\dagger} \boldsymbol{b}_k = 0 \qquad \left\{ \begin{array}{c} k > J \\ 1 \leq j \leq J \end{array} \right. , \tag{5-13}$$

the "orthogonal eigenvectors" are also orthogonal to the steering vectors. This orthogonality is exploited by the methods of this chapter.

The noise covariance structure of $\sigma^2 \boldsymbol{I}$ assumed to this point is more restrictive than required; in fact, the requirement is

- the sampled covariance matrix may be partitioned into signal and orthogonal subspaces via eigendecomposition.

This partitioning requirement implies that the signal plus noise energy is higher than the energy from noise alone—so that the J largest eigenvectors corresponds to the subspace containing the J impinging wavefronts.

In addition to permitting signal enumeration using the information-theoretic techniques of Chapter 4, the idealized noise structure results in the identified eigenvectors corresponding to an unbiased estimate of the signal subspace. Given sufficient signal strengths, the bias errors due to structured noise may be operationally negligible. If covariance whitening is applied and special array topologies are assumed (e.g., uniform linear), then the identified eigenvectors should be "de-whitened" to permit application of those algorithms.

As a final note, in the algorithms which will be described in the ensuing sections we have implicitly assumed

- the sensor array is comprised of K *identical* sensors,

or, equivalently, that the array is calibrated so that sensor variations may be removed. In addition to introducing errors into the signal enumeration and subspace partitioning, sensor response errors will result in DOA estimation errors. This is especially an issue for the polynomial-rooting implementations of MUSIC and min-Norm which assume a uniform linear array structure to permit an AR model of the wavefront propagation across the array. As the size of the array increases, relatively small perturbations in the sensor response functions can result in relatively large root (i.e. DOA) estimation errors. The importance and implications of array calibration is discussed in more detail in Section 8.4.

5.2 The MUSIC Algorithm

The MUSIC—MUltiple SIgnal Characterization[3]—algorithm was proposed by Ralph Otto Schmidt [1] in an attempt to determine whether *"the notational convenience of linear algebra with its vectors and matrices and the intuitional and analytic power afforded by vector spaces* [can] *be used to solve the practical and pressing multiple signal direction-finding problem?"* Judging by the number of papers on the MUSIC and the subsequent variations like ESPRIT, GEESE, Weighted Subspace Fitting (WSF), etc., the answer appears to be an unqualified affirmative.

Schmidt identified the orthogonality of the steering vectors demonstrated in Eqn (5-12) and proposed exploiting the orthogonality to determine the directions of arrival. The zeros of the function,

$$Q(\theta) = \sum_{k=J+1}^{K} \left|\boldsymbol{b}_k{}^{\dagger}\, \boldsymbol{d}(\theta)\right|^2 = 1 - \frac{1}{K}\sum_{k=1}^{J} \left|\boldsymbol{b}_k{}^{\dagger}\, \boldsymbol{d}(\theta)\right|^2 . \qquad (5\text{-}14)$$

correspond to the directions of arrival—where the $1/K$ scale factor is required due to our definition of an unnormalized $\boldsymbol{d}(\varphi)$ in Eqn (2-13). Alternately, the directions of arrival may be found from the peaks of the function,

$$P_{\mathrm{M}}(\theta) = \frac{1}{Q(\theta)} = \frac{1}{\boldsymbol{d}^{\dagger}(\theta)\boldsymbol{B}_o\boldsymbol{B}_o{}^{\dagger}\,\boldsymbol{d}(\theta)} \qquad (5\text{-}15)$$

which corresponds to the spatial response of the MUSIC algorithm. The normalized spatial response of the MUSIC algorithm relative to the classical methods is illustrated in Figure 3-10 for strong signals in a low-noise environment and in Figure 5-1 for the same scenario with different signal amplitudes and noise levels impinging upon a six-element array. In this set of simulated data, the information theoretic detection criteria of Chapter 4 yield the correct estimate of the number of signals and the MUSIC spatial response identifies the source directions.

[3] The MUSIC acronym is sometimes referred to as MUltiple Signal *Classification*; however, Schmidt's original definition is more accurate.

MUSIC can only tolerate *less than* K sources—otherwise, we don't have a noise subspace to which all the impinging wavefronts *must* be orthogonal. The MUSIC algorithm may be summarized as:

1) Build the sampled array covariance matrix, $\boldsymbol{R}$, from the array data;
2) Calculate the eigenvalues and eigenvectors of $\boldsymbol{R}$;
3) Estimate the number of impinging wavefronts, J (see Chapter 4); and
4) Search the spatial response function, $P_{\mathrm{M}}(\theta)$, for the extremals.

Due to the effects of finite sampling intervals as well as the other sampled covariance perturbations discussed in Chapter 8, the steering vectors, $\boldsymbol{d}(\theta)$, *will not* be *exactly* orthogonal to the eigenvectors of the *identified* orthogonal subspace—thus, we must, in general, search for the *minima* of the projection of the steering vectors onto the orthogonal subspace rather than the *zeros*. As illustrated by Figure 5-1, the depth of the null may be viewed as an indicator of the signal subspace strength and stability—which loosely corresponds to signal strength.

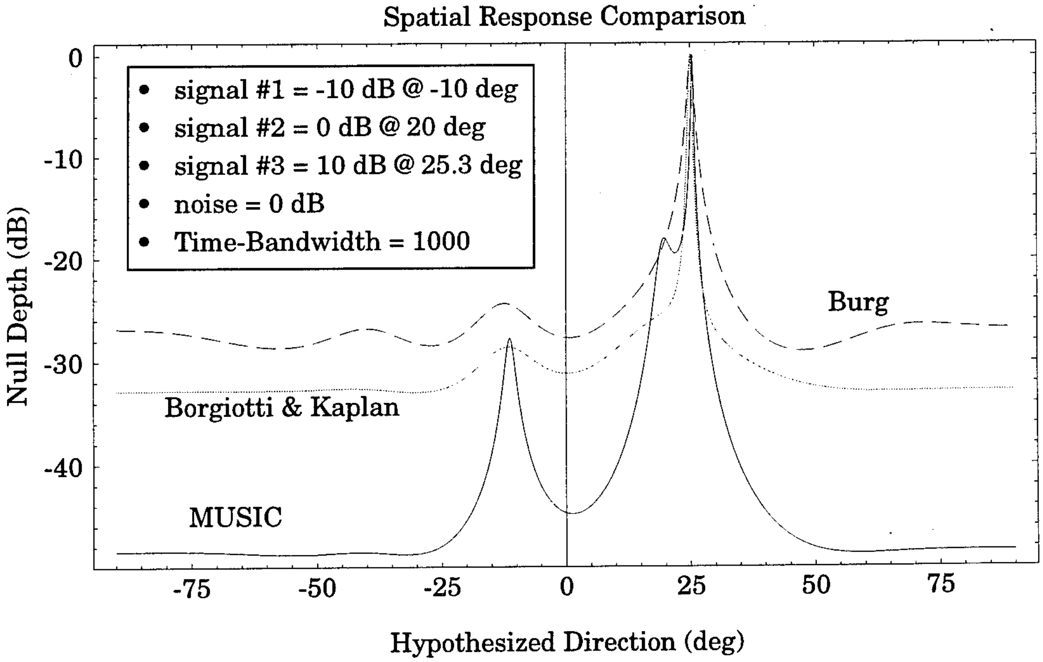


Figure 5-1: MUSIC vs. Classical Methods for Disparate Signal Strengths

The extremal search for the signal DOAs can be a computationally intensive process; to illustrate, for a six-element array approximately an order-of-magnitude more computations are required for a MATLAB implementation of the MUSIC algorithm relative to that of the GEESE algorithm (Chapter 6, page 80). However, the MUSIC algorithm is general—permitting arbitrary array topologies—as opposed to the more computationally efficient algorithms which typically require a uniform linear array structure.

5.3 Root-MUSIC

Barabell [7] recognized that if the receive array is uniform linear, the extremal search of the general MUSIC algorithm may be converted to a root-finding problem. Adopting the notation,

$$z = e^{-\imath 2\pi \delta_\lambda \sin\theta}, \tag{5-16}$$

for a uniform linear array, the steering vector, $\boldsymbol{d}(\theta)$, becomes

$$\boldsymbol{d}(\theta) = \begin{bmatrix} 1 \\ z \\ z^2 \\ \vdots \\ z^{K-1} \end{bmatrix} = \boldsymbol{d}(z). \tag{5-17}$$

In this case, the spatial response of Eqn (5-14) may be expressed in terms of matrix notation and the orthogonal subspace as [55],

$$Q(z) = \boldsymbol{d}^\dagger(z)\boldsymbol{B}_o\boldsymbol{B}_o^\dagger\,\boldsymbol{d}(z) = A^2 \prod_{k=1}^{K-1} (1 - r_k z^{-1})\,(1 - r_k^{*} z). \tag{5-18}$$

$Q(z)$ is a $2(K-1)^{\text{th}}$-order polynomial which, therefore, has $2(K-1)$ roots which occur in conjugate pairs. The roots corresponding to impinging wavefronts will—in the absence of noise-induced perturbations—lie on the unit circle since they fit the assumed propagation model. The spurious roots due to the additive noise will, in general, be dispersed away from the unit circle. Since sampled covariance perturbations will ensure that *no* roots lie *exactly* on the unit circle, normal procedure is to assume that the J root-pairs closest to the unit circle correspond to the wavefronts. Thus, the root-MUSIC algorithm may be summarized as

1) Build the sampled array covariance matrix, $\hat{\boldsymbol{R}}$

2) Perform an eigendecomposition on $\hat{\boldsymbol{R}}$

3) Estimate the number of impinging signals, $\hat{J}$

4) Construct the polynomial $Q(z)$

5) Select the $\hat{J}$ largest roots *inside* the unit circle (since roots outside the unit circle replicate the information of the roots inside the unit circle).

6) Map the selected roots into their corresponding geometric angles

Avoiding the extremal search leads to root-MUSIC being computationally more efficient than the general MUSIC algorithm. Additionally, it has a "higher resolution" since geometrically adjacent signals may be located which may not be identified during the extremal search. The root-MUSIC performance is illustrated in Figure 5-2 with the two roots closest to the unit circle corresponding to the true signal DOAs.

5.4 Pisarenko's Method

Prior to Schmidt, Pisarenko [39] proposed an algorithm for the estimation of signal frequencies which also applies to the dual problem of spatial DOA estimation. This algorithm is essentially equivalent to MUSIC algorithm under the assumption that $K-1$ signals are present. The

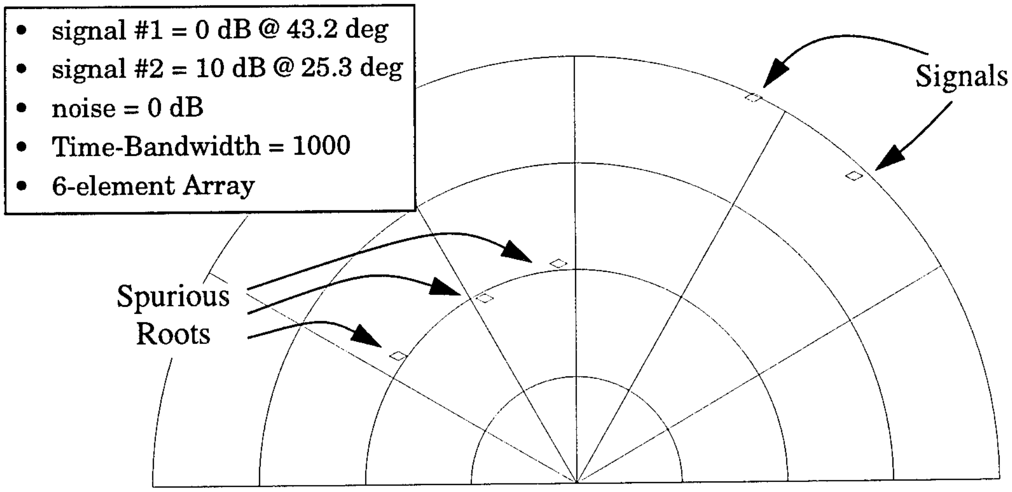


Figure 5-2: Example Root-MUSIC Root Locations

effect on the spatial spectrum response is illustrated in Figure 5-3 for simulated data. For uniform linear arrays, a root-finding approach could be adopted which solves for the roots of

$$Q(z) = \boldsymbol{d}(z)^{\dagger}\, \boldsymbol{b}_K = A \prod_{k=1}^{K-1} 1 - r_k z^{-1} = 0 \qquad (5\text{-}19)$$

which is equivalent to the MUSIC algorithm under the assumption of $K-1$ impinging signals. Expressing the roots, r_k, as

$$r_k = A_k e^{\iota\psi_k}, \qquad (5\text{-}20)$$

the DOAs may be computed as

$$\theta_k = \operatorname{asin}\left(\frac{-\psi}{2\pi\delta_\lambda}\right) \qquad (5\text{-}21)$$

where δ_λ is the inter-element spacing in units of wavelengths. Unlike the root-MUSIC approach, the roots of the $(K-1)^{\text{th}}$-order polynomial of Eqn (5-19) are not restricted to lie within the unit circle—as is illustrated in Figure 5-4. Although the roots corresponding to impinging wavefronts will lie near the unit circle, the problem of determining the number of signals as well as *which* root corresponds to the impinging wavefronts remains—a problem

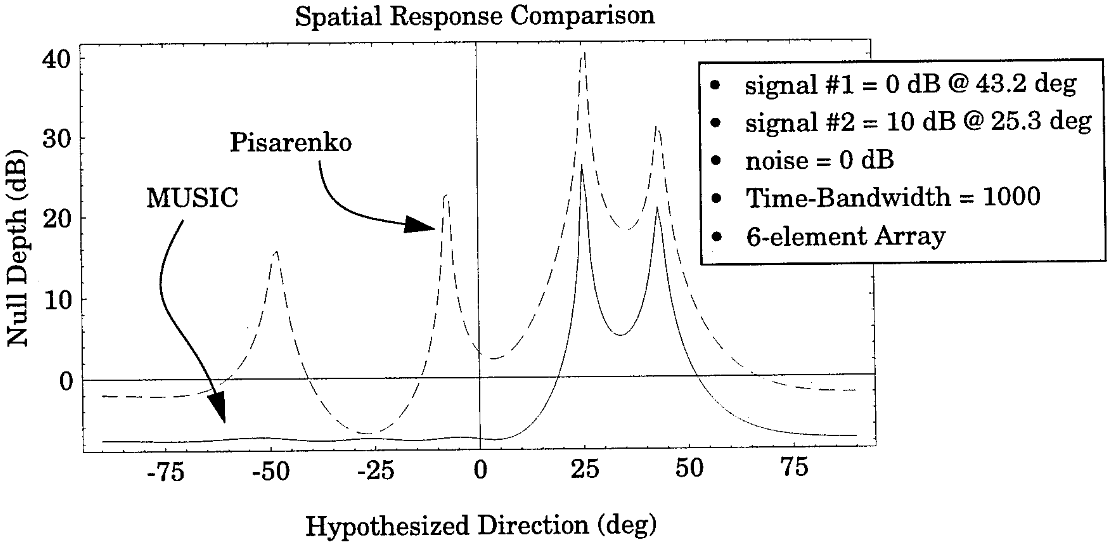


Figure 5-3: Comparison of MUSIC and Pisarenko's Method

which is made more difficult since the spurious roots may occasionally be located near the unit circle and, therefore, may appear to correspond to valid signals. Although the signal eigenvectors are constrained by the impinging wavefronts, the orthogonal eigenvectors will be randomly oriented (assuming the ideal sensors and noise model applies); hence, there is a reasonably high probability that the subspace spanned by the $K-1$ largest eigenvectors will "fit" more of the array manifold (which is derived from all possible DOAs) than required by the actually arriving signals. Unfortunately, because Pisarenko's method only exploits the smallest eigenvector, the potential of spurious roots lying near the unit circle will be independent of the SNR.

Although Pisarenko's method yields an unbiased estimate of the signal DOAs for high SNR scenarios [55], the variance of the DOA estimates will be larger than with the true subspace methods. Given the difficulty of signal detection based upon the roots of any given sampled covariance matrix, it appears that deferring the signal detection until after the DOA estimation does not yield any detection performance gains. This is consistent with Scharf's observation [66] that Pisarenko's method generally does not work very well compared to other techniques.

From an *operational* perspective, the algorithmic simplicity and computational efficiency of finding the smallest eigenvector and, thereby, detecting and locating impinging signals is attractive. Since the roots associated with the signals are fixed by the array geometry whereas

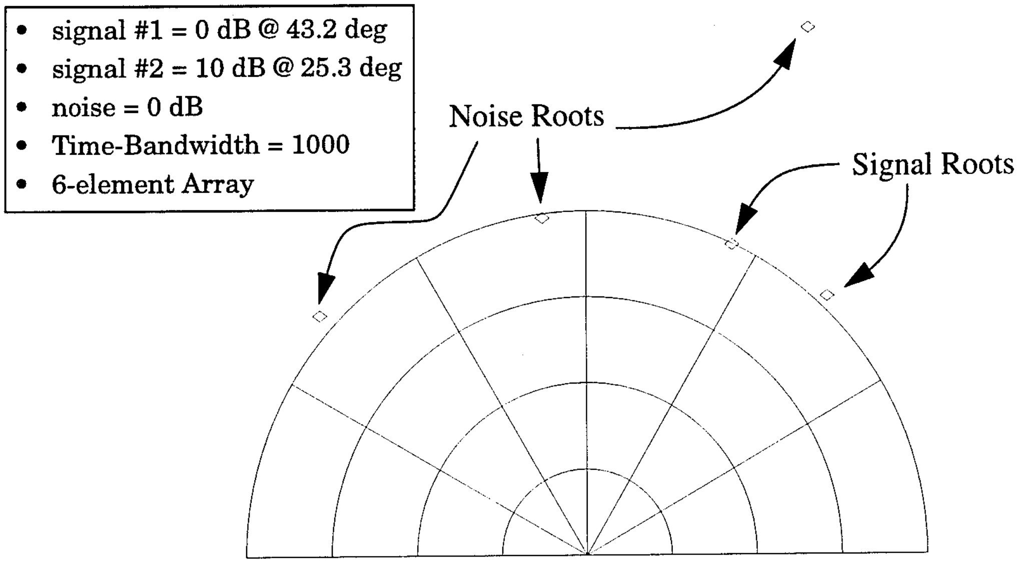


Figure 5-4: Example Root Locations using Pisarenko's Method

the spurious roots are randomly placed on the complex plane, sources may be detected and located by analyzing a (temporal) sequence of sampled covariance matrices—assuming a constant source-array geometry and continuously transmitting sources—*without* requiring *a priori* determination of the number of sources via the techniques of Chapter 4. This feature was recognized by Burkhardt [59] and was the genesis for the subspace stability exploitation methods developed in Chapter 7.

5.5 Minimum-Norm

Reddi [4] and Kumaresan and Tufts [5] proposed that the entire noise subspace be used as a reference for the polynomial rooting rather than only the smallest eigenvector as proposed by Pisarenko—the same concept used by the other subspace processing algorithms discussed in this and the following chapter. Their approach was to form a *minimum norm* vector spanning the *identified* noise subspace and to use this vector to form a polynomial—of which the roots corresponding to impinging signals would lie near the unit circle whereas the spurious roots would be uniformly distributed *inside* the unit circle in sectors not containing signals. This is illustrated in Figure 5-5. (The uniform distribution is in the complex plane, the distribution in the figure is warped by the mapping from phasor angles to geometric angles.)

Partitioning the sampled covariance eigenvectors into signal and orthogonal sets, i.e.,

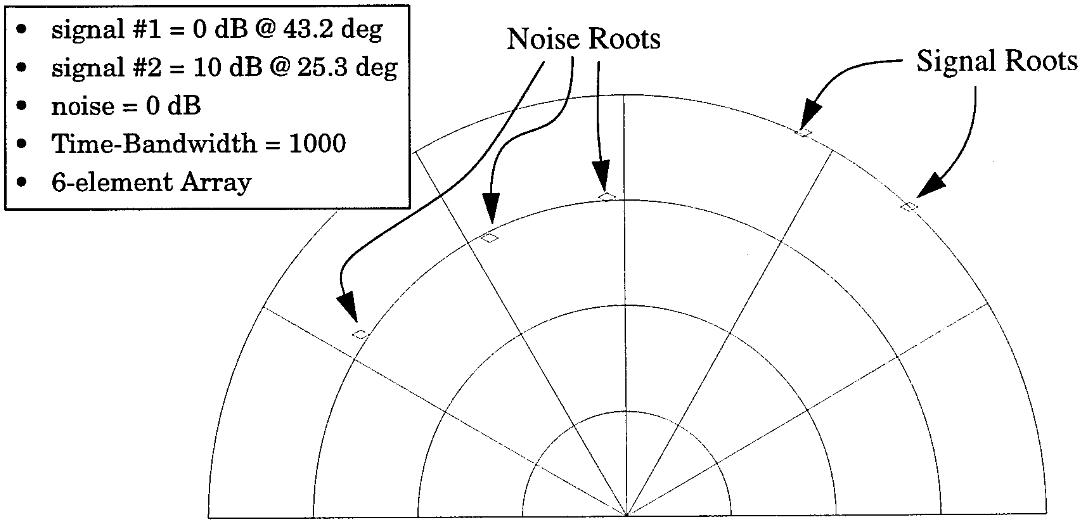


Figure 5-5: Illustration of Min-Norm Roots

$$\boldsymbol{B} = \left[\boldsymbol{B}_s\ \boldsymbol{B}_o\right], \tag{5-22}$$

we recognize that the orthogonal eigenvectors, $\boldsymbol{B}_o$, are orthogonal to the steering vectors, $\boldsymbol{d}_j(\theta_j)$ which map the impinging signals onto the array. The min-Norm algorithm proposes building a vector from the noise eigenvectors which, by definition, must also be orthogonal to the steering vectors associated with the impinging wavefronts. Define this vector, $\boldsymbol{v}$, as

$$\boldsymbol{v} = \frac{\boldsymbol{B}_0\boldsymbol{c}^\dagger}{\|\boldsymbol{c}\|} \tag{5-23}$$

where the vector, $\boldsymbol{c}$, is comprised of the first element of each of the orthogonal eigenvectors. This linear combination of the orthogonal eigenvectors results in a vector, $\boldsymbol{v}$, which is orthogonal to the signal subspace and is constrained to have an initial element of "1",

$$\boldsymbol{v}_1 = 1. \tag{5-24}$$

For arbitrary array geometries, the min-Norm algorithm may be implemented as an extremal search for the zeros of the function,

$$Q_{\mathrm{MN}}(\theta) = \left\|\boldsymbol{v}^\dagger\,\boldsymbol{d}(\theta)\right\|^2 = \boldsymbol{d}^\dagger(\theta)\boldsymbol{v}\boldsymbol{v}^\dagger\,\boldsymbol{d}(\theta), \tag{5-25}$$

or as the maxima of the spatial response,

$$P_{\mathrm{MN}}(\theta) = \frac{1}{Q_{\mathrm{MN}}(\theta)} = \frac{1}{\left\|\boldsymbol{v}^\dagger\,\boldsymbol{d}(\theta)\right\|^2} = \frac{1}{\boldsymbol{d}^\dagger(\theta)\boldsymbol{v}\boldsymbol{v}^\dagger\,\boldsymbol{d}(\theta)}. \tag{5-26}$$

For the uniform linear array, the steering vector may expressed as the auto-regressive sequence,

$$\boldsymbol{d}(\theta) = \boldsymbol{z} = \begin{bmatrix} 1 \\ z \\ z^2 \\ \vdots \\ z^{K-1} \end{bmatrix} \tag{5-27}$$

where

$$z = z(\theta) = e^{-\imath\varphi(\theta)} = e^{-\imath 2\pi\delta\sin\theta} \tag{5-28}$$

is the element-to-element phasor shifting the signal phase due to propagation across and array having an inter-element spacing of δ wavelengths. In this case, the J roots of the $(K-1)^{\text{th}}$-order polynomial,

$$\boldsymbol{v}^{\dagger}\boldsymbol{z} = 0 \tag{5-29}$$

which are on the unit circle correspond to signal roots. The spurious $K-J-1$ roots will be uniformly distributed inside the unit circle.

The root-finding implementation of the min-Norm algorithm may be summarized as:

1) build the sampled array covariance matrix, $\boldsymbol{R}$, from the data stream;
2) calculate the eigenvalues and eigenvectors of $\boldsymbol{R}$;
3) estimate the number of impinging wavefronts, J;
4) build the coefficient vector, $\boldsymbol{v}$;
5) solve the polynomial $\boldsymbol{v}^{\dagger}\boldsymbol{z} = 0$;
6) select the J roots of the polynomial closest to the unit circle; and
7) map the selected roots into their corresponding geometric angles.

Although min-Norm is slightly less accurate than either MUSIC [55],min-Norm is less computationally demanding MUSIC and may, therefore, be preferable from an implementation perspective.

5.6 Orthogonal Subspace Methods Summary

The advantages and disadvantages of the orthogonal subspace methods discussed in this chapter are summarized in Table 5-1.

For applications such as conformal arrays, the orthogonal subspace methods may be the only viable subspace processing approach. However, since the computational load is proportional to the number of elements in the sensor array, the signal subspace algorithms discussed in Chapter 6 may be preferable for uniform linear array topologies.

Additionally, the polynomial-rooting implementations of the orthogonal subspace methods assume an AR propagation model applies and, as a result, require that the sensors be identical.

Table 5-1: Advantages and Disadvantages of Orthogonal Subspace Methods

Advantages	• suitable for arbitrary array topologies • extremal search implementations are relatively robust to perturbations in the array manifold
Disadvantages	• require an accurate signal enumeration and subspace partitioning • computational load proportional to array size • polynomial-rooting implementations require accurate array calibration

Since the roots of high-order polynomials are very sensitive to perturbations in the polynomial coefficients, array calibration errors can translate into significant DOA estimation errors. This sensitivity increases as the size of the array (i.e., polynomial) increases. Array calibration is discussed in further detail in Section 8.4.

6. Signal Subspace Methods

The subspace-based Direction-of-Arrival (DOA) estimation algorithms may be characterized as either looking for signals orthogonal to the noise subspace (e.g., MUSIC) or operating directly on the signal subspace (ESPRIT, GEESE, etc.). In this chapter we explore the latter approach. The algorithms to be explored are:

- ESPRIT,
- TLS-ESPRIT, and
- GEESE.

Although there are a number of other signal subspace algorithms (e.g., PRO-ESPRIT, TAM, matrix-Pencil, etc.) all exploit the matrix-shifting nature of sensor doublets—which is the array topology common to the signals subspace methods. Since our intent in this chapter is pedagogical, we will focus on the original signal subspace algorithm, ESPRIT, and two of the subsequent variations, TLS-ESPRIT and GEESE.

Due to the array topology restrictions, the signal subspace approaches are typically much more computationally efficient than their orthogonal subspace cousins. Additionally, they relax some of the array calibration and noise covariance structure constraints associated with the orthogonal subspace algorithms.

6.1 Assumptions

The signal subspace methods assume the array is composed of *sensor doublets*—as is illustrated in Figure 6-1. Although each sensor in a doublet is assumed to have the same response, the various doublets in the array are permitted to have differing response behaviors. In this chapter we will assume the most common array topology which satisfies the doublet configuration—the uniform linear array. As illustrated in Figure 6-2, a K-element uniform sensor array may be partitioned into offset M-element subarrays which are identical except for a spatial translation, δ. In general, the subarray size should be maximized so $M = K-1$.

The signal subspace methods rely upon the assumptions that

- narrowband, uncorrelated wavefronts are impinging on the array,
- the number of wavefronts, J, is known, and
- the sampled covariance matrix may be partitioned into signal and orthogonal subspaces via eigendecomposition.

Given these assumptions, the signal received at the total array, $\boldsymbol{x}(t)$, may be expressed as

$$\boldsymbol{x}(t) = \boldsymbol{D}\boldsymbol{s}(t) + \boldsymbol{n}(t) \tag{6-1}$$

where $\boldsymbol{D}$ is the steering matrix which maps the J waveforms, $\boldsymbol{s}(t)$ onto the array and $\boldsymbol{n}(t)$ represents additive sensor noise. The signal at the two subarrays may be represented as

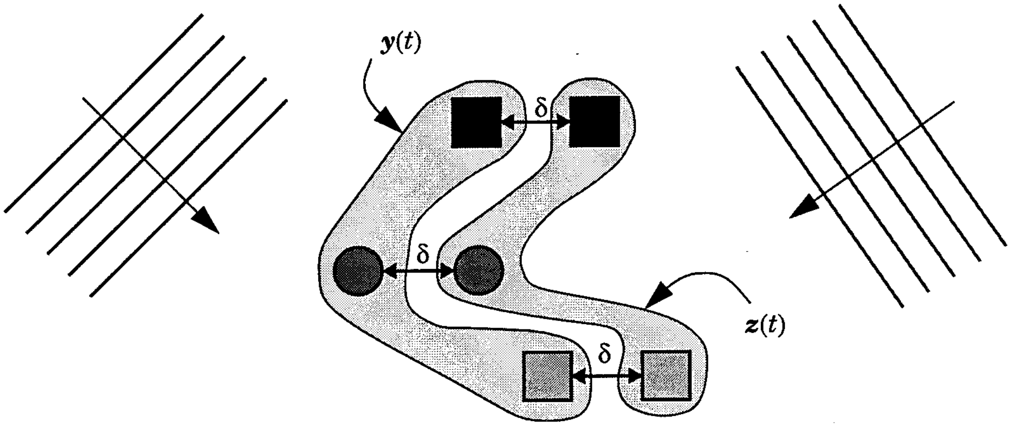


Figure 6-1: Signal Subspace Geometry Requirement of *Sensor Doublets.*

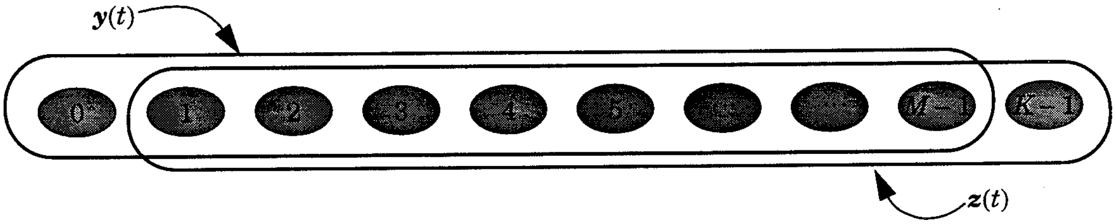


Figure 6-2: The Subarrays of a Uniform Linear Array

$$\boldsymbol{y}(t) = \tilde{\boldsymbol{D}}\boldsymbol{s}(t) + \boldsymbol{n}_y(t), \tag{6-2}$$

and,

$$\boldsymbol{z}(t) = \tilde{\boldsymbol{D}}\Phi\boldsymbol{s}(t) + \boldsymbol{n}_z(t), \tag{6-3}$$

where $\tilde{\boldsymbol{D}}$ is the steering matrix mapping signals at a reference element in the subarray onto the subarray and Φ is the $J \times J$ phase (time) shift matrix—or (in control systems terminology) the state transition matrix,

$$\Phi = \begin{bmatrix} \nu_1 & & & & & \\ & \nu_2 & & 0 & & \\ & & \nu_3 & & & \\ & & & \nu_4 & & \\ & 0 & & & \dots & \\ & & & & & \nu_J \end{bmatrix}, \tag{6-4}$$

where ν_j is the phase shift corresponding to the propagation time from the sensor element in $\boldsymbol{y}$ to the corresponding element in $\boldsymbol{z}$, i.e.,

$$\nu_j = \Phi_{jj} = e^{-\imath\varphi_j} = e^{-\imath 2\pi\delta\sin\theta_j}. \tag{6-5}$$

Here δ is the doublet spacing in units of wavelengths—which is equivalent to the inter-element spacing in this case.

The goal of the signal subspace algorithms is to determine the set of phasors, ν_j, and, thereby, determine the directions-of-arrival. The first algorithm to attempt DOA estimation by exploiting the behavior of sensor doublets was the ESPRIT algorithm.

6.2 The ESPRIT Algorithm

In their seminal paper [8], Roy, Paulraj, and Kailath recognized the implications of the translational invariance of sensor doublets and proposed the ESPRIT (Estimation of Signal Parameters via Rotational Invariance Techniques) algorithm. Therein, they proposed that the auto- and cross-covariance matrices of the two subarrays be constructed. Under an idealized (uncorrelated) noise assumption, the result is

$$\boldsymbol{R}_{yy} = E\,[\boldsymbol{y}(t)\boldsymbol{y}^{\dagger}(t)] = \boldsymbol{D}\boldsymbol{R}_s\boldsymbol{D}^{\dagger} + \sigma^2\boldsymbol{I}_M, \tag{6-6}$$

and,

$$\boldsymbol{R}_{yz} = E\,[\boldsymbol{y}(t)\boldsymbol{z}^{\dagger}(t)] = \boldsymbol{D}\boldsymbol{R}_s\Phi^{\dagger}\boldsymbol{D}^{\dagger} + \sigma^2\boldsymbol{J}_M, \tag{6-7}$$

where $\boldsymbol{I}_M$ is a $M \times M$ identity matrix and $\boldsymbol{J}_M$ is a $M \times M$ matrix with ones along the first lower diagonal off the major diagonal and zeros elsewhere, i.e.,

$$\boldsymbol{J}_M = \begin{bmatrix} 0 & 0 & 0 & \ldots & 0 & 0 \\ 1 & 0 & : & \ldots & \ldots & 0 \\ 0 & 1 & & \ldots & : & 0 \\ : & 0 & & 0 & \ldots & : \\ : & : & & 1 & 0 & 0 \\ 0 & 0 & \ldots & 0 & 1 & 0 \end{bmatrix}. \tag{6-8}$$

Notice that the assumed form for the noise contribution to the covariance matrices requires that the noise at each sensor be uncorrelated with that present at its doublet "partner" (i.e., spatially white).

As an aside, also note that the auto-covariance matrix, $\boldsymbol{R}_{yy}$, and cross-covariance matrix, $\boldsymbol{R}_{yz}$, may be extracted from the array sampled covariance matrix, $\boldsymbol{R}$,

$$\boldsymbol{R} = \boldsymbol{D}\boldsymbol{R}_s\boldsymbol{D}^{\dagger} + \Sigma_n = \boldsymbol{D}\boldsymbol{R}_s\boldsymbol{D}^{\dagger} + \sigma^2\boldsymbol{I}, \tag{6-9}$$

with $\boldsymbol{R}_{yy}$ comprising the $M \times M$ matrix ranging from the R_{11} element to the R_{MM} element and $\boldsymbol{R}_{yz}$ corresponding to the $M \times M$ matrix ranging from the R_{12} element to the R_{MK} element.

Assuming we have uncorrelated signals and uncorrelated noise, by analyzing the eigenvalues of Eqn (6-6) we can estimate $\hat{\sigma}^2$ as the average of the smallest $K-J$ eigenvalues of $\boldsymbol{R}$; this is consistent with our arguments leading to Eqn (5-7). Using this estimate, we can attempt to remove the noise covariances from the sampled data covariances to get the covariances associated with the signals, i.e.,

$$\boldsymbol{C}_{yy} = \boldsymbol{R}_{yy} - \hat{\sigma}^2\boldsymbol{I}_M \approx \boldsymbol{D}\boldsymbol{R}_s\boldsymbol{D}^{\dagger} \tag{6-10}$$

and

$$\boldsymbol{C}_{yz} = \boldsymbol{R}_{yz} - \hat{\sigma}^2 \boldsymbol{J}_M \approx \boldsymbol{D}\boldsymbol{R}_s \Phi^\dagger \boldsymbol{D}^\dagger \tag{6-11}$$

where $\boldsymbol{C}_{yy}$ and $\boldsymbol{C}_{yz}$—assuming we used a *perfectly* accurate noise model—are not of full rank. Using these approximations, we find that

$$\boldsymbol{C}_{yy} - \gamma \boldsymbol{C}_{yz} \approx \boldsymbol{D}\boldsymbol{R}_s (\boldsymbol{I}_J - \gamma \Phi^\dagger) \boldsymbol{D}^\dagger \; . \tag{6-12}$$

Since both $\boldsymbol{D}$ and $\boldsymbol{R}_{yy}$ are of rank J, the singular values of this "matrix pencil" correspond to the roots of

$$|\boldsymbol{I} - \gamma \Phi^\dagger| = 0. \tag{6-13}$$

The desired singular values may also be found as the generalized eigenvalues of the equation,

$$|\boldsymbol{C}_{yy} - \gamma \boldsymbol{C}_{yz}| = 0. \tag{6-14}$$

In the absence of noise-induced errors, the singular values are equal to the complex-conjugate of the phase shift from the sensor of $\boldsymbol{y}$ to its doublet partner in $\boldsymbol{z}$, i.e.,

$$\gamma_j \approx \Phi_{jj} = \nu_j^* = e^{\iota\varphi_j} \qquad j = 1, 2, \ldots, J. \tag{6-15}$$

From Eqn (6-15) we see that the directions of arrival can be found without resorting to the search required by the MUSIC algorithm (other than the search to find the roots of the matrix pencil)—with possibly significant savings in computation and storage requirements.

To summarize the ESPRIT algorithm, the steps involved in estimating the source directions are:

1) Define the subarrays and build the auto- and cross-covariance matrices, $\boldsymbol{R}_{yy}$ and $\boldsymbol{R}_{yz}$;

2) Estimate the noise covariance matrix $\Sigma_n \approx \hat{\sigma}^2 \boldsymbol{I}_K$ via an eigenvalue decomposition of $\boldsymbol{R}$;

3) Subtract the appropriate noise auto- and cross-covariance matrices from the auto- and cross-covariance matrices to produce the matrices, $\boldsymbol{C}_{yy}$ and $\boldsymbol{C}_{yz}$;

4) Find the singular values, γ_j, of $\boldsymbol{C}_{yy} - \gamma \boldsymbol{C}_{yz}$, i.e., find the generalized eigenvalues.

5) Translate the complex-valued singular values into a corresponding direction of arrival (simple).

Unfortunately, in practice errors due to subtracting an *estimated* noise covariance[1] from the auto- and cross-covariance matrices can lead to poor results. To alleviate these errors, the TLS-ESPRIT algorithm was developed by Roy and Kailath [9] which processes $\boldsymbol{y}$ and $\boldsymbol{z}$ simultaneously—albeit, at a cost of greater computational complexity and loading.

An additional implementation problem is that computing the generalized eigenvalues of $\boldsymbol{C}_{yy} - \gamma \boldsymbol{C}_{yz}$ will result in M singular values rather than the desired J which correspond to the impinging signals.In the noise free environment the singular values of interest will be located on the unit circle; hence, the J singular values closest to the unit circle should correspond to the signals and the spurious singular values should be near the origin. *Unfortunately,* when operating with marginal signal-to-noise ratios this criterion will sometimes fail—and produce erroneous DOA estimates. The merits and demerits of the ESPRIT algorithm are summarized in Table 6-1.

[1] If we have *a priori* knowledge of the noise covariance or can make a reasonable estimate (see Le Cadre [51]), we could handle the structured noise case. Thus if we have the estimate of the noise covariance,

$$\Sigma_n \approx \hat{\Sigma}_n,$$

then, the estimates of the sampled signal covariance matrix without the noise contribution, $\boldsymbol{C}_{yy}$ and $\boldsymbol{C}_{yz}$ could be expressed as,

$$\boldsymbol{C}_{yy} = \boldsymbol{R}_{yy} - \hat{\Sigma}_y \approx \boldsymbol{D}\boldsymbol{R}_s\boldsymbol{D}^\dagger$$

and

$$\boldsymbol{C}_{yz} = \boldsymbol{R}_{yz} - \hat{\Sigma}_z \approx \boldsymbol{D}\boldsymbol{R}_s\Phi^\dagger\, \boldsymbol{D}^\dagger$$

where the estimates of the subarray noise covariances are the appropriate submatrices of the modeled noise covariance matrix. This approach represent a generalization of the "standard" ESPRIT algorithm suitable when a reasonably accurate model of the noise covariance is available. In practice, we would generally prefer to use the GEESE algorithm discussed in Section 6.4 and avoid the need for a noise covariance model.

Table 6-1: Advantages and Disadvantages of ESPRIT

Advantages	• simple algorithm • computationally efficient • more accurate than Pisarenko's method
Disadvantages	• does not exploit the subspace partitioning • requires a high SNR • requires an accurate noise covariance model • requires a singular value selection criteria

The most serious deficiency of the ESPRIT algorithm is that it does not exploit the subspace partitioning which was implicitly used to estimate the noise covariance. As a result, practical use requires a high SNR and accurate noise covariance model. To address some of these performance concerns, Roy and Kailath extended the ESPRIT concept and developed the TLS-ESPRIT algorithm discussed in the following section.

6.3 TLS-ESPRIT

The TLS-ESPRIT (Total Least-Squares ESPRIT) algorithm was developed by Roy and Kailath [9] to alleviate some of the problems with the ESPRIT algorithm. Use the definitions of $\boldsymbol{y}$ and $\boldsymbol{z}$ and build the supervector,

$$\tilde{\boldsymbol{x}} = \begin{bmatrix} \boldsymbol{y} \\ \boldsymbol{z} \end{bmatrix} = \begin{bmatrix} \boldsymbol{D} \\ \Phi\boldsymbol{D} \end{bmatrix} \boldsymbol{s}(t) + \begin{bmatrix} \boldsymbol{n}_y \\ \boldsymbol{n}_z \end{bmatrix} = \tilde{\boldsymbol{D}}\boldsymbol{s} + \tilde{\boldsymbol{n}}, \tag{6-16}$$

where $\tilde{\boldsymbol{D}}$ is a $2M \times J$ matrix as implicitly defined by Eqn (6-16). If we compute the covariance of this supervector, we see that

$$\boldsymbol{R}_{\tilde{x}} = E\left[\tilde{\boldsymbol{x}}\tilde{\boldsymbol{x}}^{\dagger}\right] = \tilde{\boldsymbol{D}}\boldsymbol{R}_s\tilde{\boldsymbol{D}}^{\dagger} + \boldsymbol{R}_{\tilde{n}} \tag{6-17}$$

where, as with our ESPRIT development, if we assume we have a uniform linear array (illustrated by Figure 6-2) so that our two subarrays are overlapping. Assuming the noise is spatially uncorrelated, the noise covariance matrix has the form,

$$\boldsymbol{R}_{\tilde{n}} = \sigma^2\Sigma_{\tilde{\boldsymbol{n}}} = \sigma^2\begin{bmatrix} \boldsymbol{I}_M & \boldsymbol{J}_M \\ \boldsymbol{J}_M{}^{\dagger} & \boldsymbol{I}_M \end{bmatrix} \tag{6-18}$$

where σ^2 is the sensor noise power. Here $M = K-1$ is the size of the subarrays used to generate $\boldsymbol{y}$ and $\boldsymbol{z}$, and $\boldsymbol{I}_M$ is the $M \times M$ identity matrix. Since $\boldsymbol{R}_s$ is of rank J then the product $\tilde{\boldsymbol{D}}\boldsymbol{R}_s\tilde{\boldsymbol{D}}^{\dagger}$ is also of rank J; this implies that the generalized eigenvalues[2] of $\boldsymbol{R}_z$ can be represented as an ordered set

$$\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_J > \lambda_{J+1} = \ldots = \lambda_{2M} = \sigma^2. \tag{6-19}$$

Similar to the MUSIC algorithm logic [see Eqn (5-11)–Eqn (5-14)], the eigenvalues and eigenvectors associated with the noise don't have a contribution from the signals, hence,

$$\boldsymbol{R}_{\tilde{x}}\boldsymbol{e}_i = \lambda_i \boldsymbol{R}_{\tilde{n}}\boldsymbol{e}_i \qquad i = J+1, J+2, \ldots, 2M, \tag{6-20}$$

where $\boldsymbol{e}_i$ represents the $2M$ element eigenvector corresponding to the eigenvalue λ_i. Therefore, the noise eigenvectors are orthogonal to the (super-) steering vectors which map the signal phase onto the receive array, i.e.,

$$\tilde{\boldsymbol{d}}_j^{\dagger}\,\boldsymbol{e}_i = 0 \qquad j = 1, 2, \ldots, J; \qquad k = J+1, \ldots, 2M. \tag{6-21}$$

This, in turn, implies that the signal eigenvectors span the same subspace spanned by the column vectors of $\tilde{\boldsymbol{D}}$. Hence, there is some *nonsingular* $J \times J$ matrix $\boldsymbol{C}$ which relates the signals' eigenvectors to the steering vectors, i.e.,

$$\left[\boldsymbol{e}_1\ \boldsymbol{e}_2\ \cdots\ \boldsymbol{e}_J\right] = \tilde{\boldsymbol{D}}\boldsymbol{C}. \tag{6-22}$$

Partition this $2M \times J$ eigenvector matrix into two $M \times J$ submatrices $\boldsymbol{E}_y$ and $\boldsymbol{E}_z$ so that,

$$\left[\boldsymbol{e}_1\ \boldsymbol{e}_2\ \cdots\ \boldsymbol{e}_J\right] = \begin{bmatrix}\boldsymbol{E}_y \\ \boldsymbol{E}_z\end{bmatrix}. \tag{6-23}$$

Upon appropriate substitution, this leads to,

$$\boldsymbol{E}_y = \boldsymbol{D}\boldsymbol{C} \qquad \text{and} \qquad \boldsymbol{E}_z = \boldsymbol{D}\boldsymbol{\Phi}\boldsymbol{C}. \tag{6-24}$$

Using these two matrices to construct a matrix wherein they are placed side-by-side rather than stacked, i.e.,

[2] Generalized eigenvalues are similar to the standard eigenvalue problem of $\boldsymbol{Ax} = \lambda\boldsymbol{x}$ except that the applicable equation is of the form $\boldsymbol{Ax} = \lambda\boldsymbol{Bx}$. In this case, the equation is $\boldsymbol{R}_x\boldsymbol{e} = \lambda\boldsymbol{R}_{\tilde{n}}\boldsymbol{e}$.

$$\left[\boldsymbol{E}_y\ \boldsymbol{E}_z\right] = \boldsymbol{D}\left[\boldsymbol{C}\ \Phi\boldsymbol{C}\right], \tag{6-25}$$

then we can define yet another matrix which is the "square" (and of size $2J \times 2J$) of this matrix,

$$\boldsymbol{E}_{yz} = \begin{bmatrix} \boldsymbol{E}_y{}^\dagger \\ \boldsymbol{E}_z{}^\dagger \end{bmatrix} \left[\boldsymbol{E}_y\ \boldsymbol{E}_z\right] = \begin{bmatrix} \boldsymbol{C}^\dagger \\ \boldsymbol{C}^\dagger\ \Phi^\dagger \end{bmatrix} \boldsymbol{D}^\dagger\, \boldsymbol{D}\left[\boldsymbol{C}\ \Phi\boldsymbol{C}\right]. \tag{6-26}$$

Since $\boldsymbol{E}_{yz}$ is nonnegative-definite Hermitian (due to the complex-conjugate multiplication used in its definition and it not being of full rank) and is of rank J (since $\boldsymbol{D}^\dagger\, \boldsymbol{D}$ is of rank J), it may be expressed as,

$$\boldsymbol{E}_{yz} = \boldsymbol{V} \begin{bmatrix} l_1 & & & & & \\ & l_2 & & \boldsymbol{0} & & \\ & & \cdots & & & \\ & & & l_J & & \\ & & & & 0 & \\ & \boldsymbol{0} & & & & \cdots \\ & & & & & 0 \end{bmatrix} \boldsymbol{V}^\dagger \tag{6-27}$$

where $l_j > 0$ and $\boldsymbol{V}\boldsymbol{V}^\dagger = \boldsymbol{I}_M$[3]. For later use, we partition $\boldsymbol{V}$ into $J \times J$ submatrices, i.e.,

$$\boldsymbol{V} = \begin{bmatrix} \boldsymbol{V}_{11} & \boldsymbol{V}_{12} \\ \boldsymbol{V}_{21} & \boldsymbol{V}_{22} \end{bmatrix}. \tag{6-28}$$

The next step is to find a $2J \times J$ full rank matrix $\boldsymbol{W}$ such that

$$\left[\boldsymbol{E}_y\ \boldsymbol{E}_z\right]\boldsymbol{W} = 0. \tag{6-29}$$

Substituting in from Eqn (6-25) leads to the equivalent expression,

$$\boldsymbol{D}\left[\boldsymbol{C}\ \Phi\boldsymbol{C}\right]\boldsymbol{W} = 0. \tag{6-30}$$

[3] The $\boldsymbol{V}$ is an orthonormal matrix so this equation essentially amounts to a *similarity transformation* on the matrix $\boldsymbol{E}_{yz}$. The task of finding the values l_j and the matrix $\boldsymbol{V}$ corresponds to determining the eigenvalues and eigenvectors of $\boldsymbol{E}_{yz}$.

Since $\boldsymbol{D}$ is a $M \times J$ matrix and is of rank J, for the above equation to be satisfied, we must have,

$$\begin{bmatrix}\boldsymbol{C} & \Phi\boldsymbol{C}\end{bmatrix}\boldsymbol{W} = 0. \tag{6-31}$$

Partition $\boldsymbol{W}$ into two $J \times J$ matrices so that,

$$\boldsymbol{W} = \begin{bmatrix}\boldsymbol{W}_1 \\ \boldsymbol{W}_2\end{bmatrix}; \tag{6-32}$$

and substitute this into Eqn (6-31) to produce

$$\begin{bmatrix}\boldsymbol{C} & \Phi\boldsymbol{C}\end{bmatrix}\begin{bmatrix}\boldsymbol{W}_1 \\ \boldsymbol{W}_2\end{bmatrix} = \boldsymbol{C}\boldsymbol{W}_1 + \Phi\boldsymbol{C}\boldsymbol{W}_2 = 0. \tag{6-33}$$

For Eqn (6-33) to be satisfied, we must have,

$$-\boldsymbol{W}_1\boldsymbol{W}_2^{-1} = \boldsymbol{C}^{-1}\Phi\boldsymbol{C}. \tag{6-34}$$

From Eqn (6-34) we see that the eigenvalues of the matrix $-\boldsymbol{W}_1\boldsymbol{W}_2^{-1}$ are equal to the eigenvalues of Φ[4]—i.e., $e^{-\iota\varphi_j}$ for $j = 1, 2, \ldots, J$—so that the angles-of-arrival of the incoming signals are, once again, determined directly without a search over the possible mapping vectors. To find a matrix $\boldsymbol{W}$ which satisfies the requirements of Eqn (6-29), inspect Eqn (6-27) to see that

$$\boldsymbol{E}_{yz}\boldsymbol{v}_i = l_i\boldsymbol{v}_i = 0 \qquad J < i \le 2J, \tag{6-35}$$

where $\boldsymbol{v}_i$ represents the i^{th} column vector of $\boldsymbol{V}$. Since $\boldsymbol{E}_{yz}$ is of rank J, this implies that $\begin{bmatrix}\boldsymbol{E}_y & \boldsymbol{E}_z\end{bmatrix}$ is also of rank J. To achieve the form of Eqn (6-27), we must have the equivalent representation,

$$\begin{bmatrix}\boldsymbol{E}_y & \boldsymbol{E}_z\end{bmatrix}\boldsymbol{v}_i = 0 \qquad J < i \le 2J. \tag{6-36}$$

Thus, the desired matrix $\boldsymbol{W}$ is given by

[4] The eigenvalues are the same since eigenvalues are not changed by a similarity transform—i.e., Φ has the same eigenvalues as $\boldsymbol{C}^{-1}\Phi\boldsymbol{C}$.

$$W = \left[v_{J+1}\ v_{J+2}\ v_{J+3}\ \cdots\ v_{2J}\right] \equiv \begin{bmatrix} V_{12} \\ V_{22} \end{bmatrix} \tag{6-37}$$

so that the eigenvalues of $-V_{12}V_{22}^{-1}$ yield the actual directions-of-arrival! It should be noted that the TLS-ESPRIT has better performance than that of the ESPRIT algorithm; however, this improvement is garnered at a cost of greater computational complexity. To see this, contrast the ESPRIT algorithm with the sequence of operations required by the TLS-ESPRIT algorithm:

1) Estimate the super-covariance matrix, $\hat{R}_{\tilde{x}}$, from the sampled data.
2) Estimate the noise covariance matrix structure, $R_{\tilde{n}}$.
3) Compute the generalized eigenvalues *and* eigenvectors of the matrix pencil $\{R_{\tilde{x}}, R_{\tilde{n}}\}$. (Note that this is a $2M \times 2M$ matrix.)
4) Estimate the number of sources, $\hat{J}$.
5) Build the $2J \times 2J$ matrix E_{yz} and compute the associated generalized eigenvectors, V.
6) Invert V_{22} and form the $J \times J$ matrix $\Psi = -V_{12}V_{22}^{-1}$.
7) Determine the eigenvalues of Ψ—these map directly into the directions-of-arrival of the incoming signals.

In contrast to the ESPRIT algorithm which required one eigenvalue and one generalized eigenvalue decomposition of $M \times M$ matrices, the TLS-ESPRIT approach requires a $2M \times 2M$ generalized eigenvector decomposition, a $2J \times 2J$ eigenvector decomposition, a $J \times J$ matrix inversion, and a $J \times J$ eigenvalue decomposition. In sum, the more robust TLS-ESPRIT algorithm is much more computationally demanding than the simple ESPRIT approach—although, for small arrays and fast processors, this may not be a significant concern. The merits of the TLS-ESPRIT algorithm are summarized in Table 6-2.

While the TLS-ESPRIT algorithm alleviates the ESPRIT need for a singular value selection criteria due to producing the same number of DOAs as hypothesized signals, it shares the ESPRIT deficiency of requiring an accurate noise covariance model. Neither ESPRIT algorithm exploits the subspace partitioning which was implicitly used to estimate the number of impinging wavefronts. Furthermore, the TLS-ESPRIT performance is garnered at the cost

Table 6-2: Advantages and Disadvantages of TLS-ESPRIT

Advantages	• eliminates the need for a singular value selection criteria • more accurate than ESPRIT
Disadvantages	• computationally demanding • requires an accurate noise covariance model • requires an accurate estimate of the number of impinging wavefronts

of increased algorithmic complexity and computational loading. These deficiencies are addressed by the GEESE algorithm which is the topic of the following section.

6.4 GEESE

The GEESE (GEneralized Eigenvalues utilizing Signal Subspace Eigenvectors) algorithm developed by Kwon [3] has performance comparable to the TLS-ESPRIT while also featuring computational simplicity. This is done by noting that since the signal direction vectors, $\boldsymbol{d}_j$, are orthogonal to the orthogonal subspace eigenvectors, the subspace spanned by the true direction vectors is the same as the one spanned by the eigenvectors corresponding to signals. Hence, the signal eigenvectors can be expressed as a linear combination of the direction vectors, i.e.,

$$\boldsymbol{b}_i = \sum_{j=1}^{J} \tilde{c}_{ij}\boldsymbol{d}_j \qquad i = 1, 2, \ldots, J. \tag{6-38}$$

If we form a $K \times J$ matrix from the signals' eigenvectors (recall that there are K sensors in the array and there are assumed to be J signal sources being received), then we can define

$$\tilde{\boldsymbol{B}} \equiv \left[\boldsymbol{b}_1\ \boldsymbol{b}_2\ \boldsymbol{b}_3\ \ldots\ \boldsymbol{b}_J\right]. \tag{6-39}$$

Using Eqn (6-38) leads to the representation,

$$\tilde{\boldsymbol{B}} = \boldsymbol{D}\tilde{\boldsymbol{C}} \tag{6-40}$$

where $\boldsymbol{D}$ is the steering matrix defined by Eqn (2-12) which maps the signal onto the array and $\tilde{\boldsymbol{C}}$ is a $J \times J$ nonsingular matrix. Now define two matrices $\tilde{\boldsymbol{B}}_1$ and $\tilde{\boldsymbol{B}}_2$ using the first M rows and the 2nd through $(M+1)^{\text{th}}$ rows of $\tilde{\boldsymbol{B}}$, respectively, where $J \le M \le K-1$ (i.e., M is at least equal to the number of signal sources but less than the number of sensors in the array). Thus,

$$\tilde{\boldsymbol{B}}_1 = \left[\boldsymbol{I}_M \; 0_{M,K-M}\right]\tilde{\boldsymbol{B}} \tag{6-41}$$

and

$$\tilde{\boldsymbol{B}}_2 = \left[0_{M,1} \; \boldsymbol{I}_M \; 0_{M,K-M-1}\right]\tilde{\boldsymbol{B}}, \tag{6-42}$$

where $0_{M,P}$ denotes a $M \times P$ matrix of zeros. If we recall our definition of the element-to-element phase shift for the j^{th} signal as

$$\nu_j = e^{-\imath\varphi_j} \tag{6-43}$$

then, the steering matrix for a uniform linear array is,

$$\boldsymbol{D} = \begin{bmatrix} 1 & 1 & \dots & 1 \\ \nu_1 & \nu_2 & \dots & \nu_J \\ : & : & & : \\ \nu_1^K & \nu_2^K & \dots & \nu_J^K \end{bmatrix}. \tag{6-44}$$

Furthermore, if we define two submatrices of $\boldsymbol{D}$ analogous to $\tilde{\boldsymbol{B}}_1$ and $\tilde{\boldsymbol{B}}_2$, i.e.,

$$\boldsymbol{D}_1 = \begin{bmatrix} 1 & 1 & \dots & 1 \\ \nu_1 & \nu_2 & \dots & \nu_J \\ : & : & & : \\ \nu_1^{M-1} & \nu_2^{M-1} & \dots & \nu_J^{M-1} \end{bmatrix} \tag{6-45}$$

and

$$\boldsymbol{D}_2 = \begin{bmatrix} \nu_1 & \nu_2 & \dots & \nu_J \\ \nu_1^2 & \nu_2^2 & \dots & \nu_J^2 \\ : & : & & : \\ \nu_1^M & \nu_2^M & \dots & \nu_J^M \end{bmatrix} = \Phi\boldsymbol{D}_1 \tag{6-46}$$

where, as in Eqn (6-4), Φ is

$$\Phi = \begin{bmatrix} \nu_1 & & & \\ & \nu_2 & 0 & \\ & 0 & \dots & \\ & & & \nu_J \end{bmatrix}, \tag{6-47}$$

then we see that

$$\tilde{B}_1 = D_1\tilde{C} \qquad \text{and} \qquad \tilde{B}_2 = D_2\tilde{C}. \tag{6-48}$$

Finally, let us note that

$$\tilde{B}_1 - \gamma\tilde{B}_2 = D_1\tilde{C} - \gamma D_1\Phi\tilde{C} = D_1(I_J - \gamma\Phi)\tilde{C}. \tag{6-49}$$

Thus, similar to the ESPRIT algorithm, the generalized singular values, γ_j, associated with this matrix pencil map directly into the j^{th} signal direction. To verify this, note that since the J columns of $\tilde{B}$ are independent and the number of signals is less than the number of elements in the array (i.e., $J < K$), then $\tilde{B}$ is of rank J. From the definitions of $\tilde{B}_1$ and $\tilde{B}_2$ they must also be of rank J. This in turn implies that D_1 and $\tilde{C}$ are also of rank J since $M \geq J$, by definition. Thus, the singular values of the matrix pencil $\tilde{B}_1 - \gamma\tilde{B}_2$ are given by the roots of

$$|I_J - \gamma\Phi| = 0. \tag{6-50}$$

which leads us to conclude that these generalized singular values are the complex conjugate of the diagonal elements of Φ, i.e.,

$$\Phi_{jj} = \gamma_j^* = e^{\iota\varphi_j} \qquad j = 1, 2, \dots, J. \tag{6-51}$$

Note that M can be any integer between J (the number of sources) and $K-1$ (where K is the number of sensors). Using a larger M will generally give a better estimate of the source direction since more information is used in the estimation in that case. In general, using $M = K-1$ leads to the GEESE algorithm giving better performance than the MUSIC algorithm. We should also note that $\tilde{B}_1$ and $\tilde{B}_2$ are not square matrices.

To summarize the GEESE algorithm, the steps involved in the detection and estimation of the DOAs are:

1) Build the covariance matrix of all the sensors, $\boldsymbol{R}$;

2) Do an eigendecomposition of $\boldsymbol{R}$, i.e., $\boldsymbol{R} = \sum_{k=1}^{K} \lambda_k \boldsymbol{b}_k \boldsymbol{b}_k^{\dagger}$;

3) Estimate the number of sources, $\hat{J}$;

4) Build the $K \times \hat{J}$ matrix $\tilde{\boldsymbol{B}}$ from the signal subspace eigenvectors;

5) Select the $M \times \hat{J}$ submatrices $\tilde{\boldsymbol{B}}_1$ and $\tilde{\boldsymbol{B}}_2$ from $\boldsymbol{B}$;

6) Find the generalized singular values of the matrix pencil $\tilde{\boldsymbol{B}}_1 - \gamma\tilde{\boldsymbol{B}}_2$, γ_j; and

7) Map these singular values directly into estimates of the directions of arrival.

In examining the above sequence, we see that it is quite simple since the primary computational demands are a single eigendecomposition plus the determination of the singular values of the matrix pencil.

As an aside, we should note that the generalized singular values of the non-square system $|\tilde{\boldsymbol{B}}_1 - \gamma\tilde{\boldsymbol{B}}_2| = 0$ may be found by finding the generalized eigenvalues of the equivalent system,

$$|\tilde{\boldsymbol{B}}_1 - \gamma\tilde{\boldsymbol{B}}_2| = \left|\tilde{\boldsymbol{B}}_1^{\dagger}(\tilde{\boldsymbol{B}}_1 - \gamma\tilde{\boldsymbol{B}}_2)\right| = \left|\tilde{\boldsymbol{B}}_1^{\dagger}\tilde{\boldsymbol{B}}_1 - \gamma\tilde{\boldsymbol{B}}_1^{\dagger}\tilde{\boldsymbol{B}}_2\right| = 0. \tag{6-52}$$

The two matrices, $\tilde{\boldsymbol{B}}_1^{\dagger}\tilde{\boldsymbol{B}}_1$ and $\tilde{\boldsymbol{B}}_1^{\dagger}\tilde{\boldsymbol{B}}_2$, are $J \times J$ and so are readily handled by available algorithms for calculating the generalized eigenvalues (e.g., MATLAB). Alternately, at some numerical instability risk, the roots may be found as the eigenvalues of,

$$|\tilde{\boldsymbol{B}}_1 - \gamma\tilde{\boldsymbol{B}}_2| = \left|\boldsymbol{I} - \gamma(\tilde{\boldsymbol{B}}_1^{\dagger}\tilde{\boldsymbol{B}}_1)^{-1}\tilde{\boldsymbol{B}}_1^{\dagger}\tilde{\boldsymbol{B}}_2\right| = |\boldsymbol{I} - \gamma\tilde{\boldsymbol{B}}_1^{\#}\tilde{\boldsymbol{B}}_2| = 0 \tag{6-53}$$

where $\tilde{\boldsymbol{B}}_1^{\#} = (\tilde{\boldsymbol{B}}_1^{\dagger}\tilde{\boldsymbol{B}}_1)^{-1}\tilde{\boldsymbol{B}}_1^{\dagger}$ is the pseudo-inverse of $\tilde{\boldsymbol{B}}_1$. Note that the computational load required to determine the singular values is a function of the number of estimated wavefronts, J, rather than the size of the array—this results in a significant computational advantages relative to the root-MUSIC and min-Norm algorithms as well as ESPRIT and TLS-ESPRIT.

The GEESE algorithm results in *exactly* J singular values—so that the selection criterion used by the ESPRIT algorithm to choose singular values may be avoided. The advantages and disadvantages of the GEESE algorithm are summarized in Table 6-3.

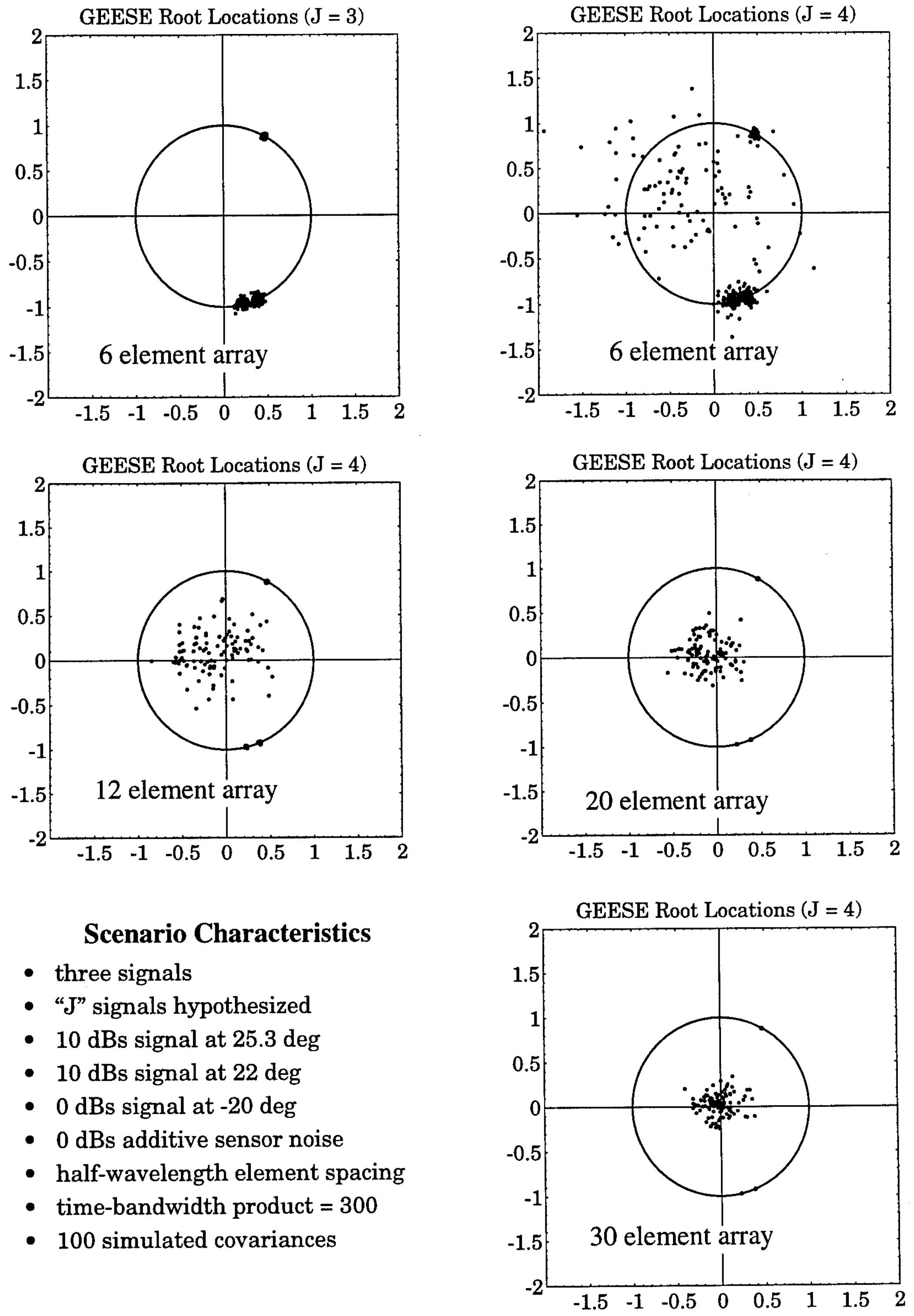
Table 6-3: Advantages and Disadvantages of GEESE

Advantages	• computationally very efficient • exploits the identified signal subspace • better accuracy than MUSIC, ESPRIT, or TLS-ESPRIT • resistant to perturbations in the array manifold • does not require a noise covariance model
Disadvantages	• requires an accurate subspace partitioning • requires an accurate signal enumeration

Whereas the ESPRIT algorithms exploit the translational invariance of the sampled signal covariance matrix, the GEESE algorithm exploits the invariance *and* the subspace partitioning of the sampled covariance. This exploitation results in the GEESE algorithm being more accurate and robust than ESPRIT or TLS-ESPRIT. Furthermore, using the identified signal subspace eliminates the vulnerability to noise covariance modelling errors. Of course, an accurate signal enumeration and subspace partitioning is required.

The GEESE algorithm may be viewed as finding the set of complex numbers (phasors) which best map the signal subspace received at one subarray into that of the translated subarray. As a result, the GEESE algorithm is more resilient to array manifold perturbations and calibration errors than the root-finding implementations of the orthogonal subspace methods. This feature coupled with the computational efficiency makes GEESE a excellent choice for real-time implementation of subspace processing.

A feature which has not been noted in the published literature is that while roots corresponding to impinging signals will reside on the unit circle, if the number of signals is overestimated the spurious roots will tend to be randomly located inside the unit circle and away from the signal root locations. This behavior illustrated in Figure 6-3 for a variety of array sizes and simulated data under a signal enumeration hypothesis one greater than the actual number of signals. Note that the spurious root offset from the unit circle tends to increase as the array size increases—which is reasonable since as the array size increases the GEESE algorithm has greater difficulty finding a singular value which will align the two identified subspaces. Although not exploited in previous applications, the SSET subspace stability method (introduced in Section 7.6) uses this GEESE characteristic in its processing. The spurious root behavior is illustrated in Figure 7-17 for in-water data in conjunction with the SSET development.



Scenario Characteristics

- three signals
- "J" signals hypothesized
- 10 dBs signal at 25.3 deg
- 10 dBs signal at 22 deg
- 0 dBs signal at -20 deg
- 0 dBs additive sensor noise
- half-wavelength element spacing
- time-bandwidth product = 300
- 100 simulated covariances

Figure 6-3: GEESE Roots vs. Enumeration Accuracy and Array Size

7. Subspace Stability Methods

Determining the number of impinging wavefronts is the most critical and difficult task associated with subspace processing since the fundamental assumptions of perfect noise,[1] perfect sensors,[2] and perfect signals[3] are frequently violated—as is discussed in Chapter 8. Given an accurate estimate of the number of signals and an accurate partitioning of the sampled covariance matrix into signal and orthogonal subspaces, the subspace-based DOA estimation algorithms yield functionally equivalent performance for most practical applications. Since the information-theoretic enumeration techniques fail in many real-world environments, we seek enumeration (and DOA estimation) algorithms which are less restrictive in their operational applicability. In this chapter we explore three techniques which exploit the *subspace stability* of continuously transmitting signal sources: *Burkhardt's Method*, the *root-Tracker* method, and the *Subspace Stability Exploitation Tracker* (SSET).

7.1 Subspace Stability Methods Overview

If signal sources transmit continuously over a "long" period of time and the source-array geometry is stable over that interval, there will be a temporal stability in the signal subspace; under the traditional assumptions of *i.i.d.* additive Gaussian noise and narrowband (near-cisoid) signals and matched sensors, there will be a corresponding *instability* of the eigenvectors within the orthogonal (noise) subspace. This chapter explores algorithms which exploit these characteristics of the two subspaces to improve the detection performance of the subspace algorithms.

Burkhardt [59] proposed to avoid the signal detection difficulties associated with traditional subspace processing by applying the spectral response form of Pisarenko's method (Section 5.4) to *sequences* of sampled covariance matrices. Given continuously transmitting signal sources,

[1] Recall that the noise is typically assumed to be equi-powered spatially-independent additive. This assumption can be violated by flow noise or multipath effects—i.e., colored noise.

[2] The sensor array elements are generally assumed to have identical gain and phase behaviors as well as precisely known placement—i.e., matched sensors.

[3] A fundamental assumption is that the impinging wavefronts are uncorrelated narrowband (cisoid) signals so the effect of wavefront propagation across the array can be modelled as a phase shift. Furthermore, the sources are assumed to be stationary point-sources over the averaging interval.

the peaks (nulls) in the spatial spectrum response derived from the signals would be stable whereas the spurious peaks due to the noise would vary randomly; eventually, sources could be identified due to their persistence with closely spaced sources recognized (albeit, not resolved) using the width of the averaged response function.

Although **Burkhardt's method** avoids the requirement for an *a priori* determination of the number of signals, it is computationally intensive since the spatial response for each hypothesized DOA must be determined for each covariance matrix in the sequence; furthermore, the spatial resolution was limited to that of the hypothesized DOA grid. The root-Tracker and SSET approaches extend Burkhardt's concepts and offer improved computational efficiency and accuracy performance.

The root-finding version of Pisarenko's method produces a set of roots—*possibly some* of which correspond to arriving wavefronts. To identify roots corresponding to signals, the **root-Tracker** applies multi-target tracking techniques to the *sequence* of complex-valued roots derived from a series of sampled covariance matrices; the identification assumes signal roots are stable and reside on the unit circle—features which are not associated with the spurious roots derived from the additive noise.

The **SSET** is similar to the root-Tracker except that by melding *multiple-hypothesis tracking* techniques and the GEESE signal subspace algorithm improved computational and accuracy performance may be achieved. The SSET is a computationally efficient algorithm with a demonstrable ability to process acoustical array data against which conventional subspace methods fail—as is shown in Section 7.6.3.

7.2 Introduction

The span and orientation of the hyperplane representing the signal subspace is a function of the array topology and the source-array geometry; if the geometry is static, the hyperplane will be constant across a sequence of covariance matrices. Consequently, the orthogonal subspace will—by definition—also be constant. Figure 7-1 illustrates the signal hyperplane and the additive noise "ball". The classical subspace processing assumption is that the noise is spherically symmetric so that there is no preferred orientation—in such a case, the noise eigenvalues would be equal. In practice, sampling effects will guarantee that the noise "ball" is perturbed from this ideal structure so that there is a preferred—albeit random between sampled covariance matrices—orientation. Note that the additive noise spans the entire space so that the common designation of the *orthogonal* (non-signal) subspace as the "*noise*" subspace is, in fact, a misnomer. However, we shall use the two terms interchangeably.

Ideally, we would like to detect and estimate impinging signals using all of the available information concerning the array manifold (i.e., how the signal wavefront maps onto the array elements), signal strengths, noise structure, propagation effects, etc.; unfortunately, the computational implications of such a multi-dimensional optimization problem generally makes such an approach impractical. A more tractable (suboptimal) approach is to sequence the signal enumeration and characterization by making the fundamental assumptions:

- the array response is perfectly known,
- the additive noise covariance structure is perfectly known,
- the signals are narrowband, and
- the signals and noise are uncorrelated.

Under the last two assumptions, the received signal vector, $\boldsymbol{x}(t)$, may be modelled as

$$\boldsymbol{x}(t) = \boldsymbol{D}(\theta)\boldsymbol{s}(t) + \boldsymbol{n}(t) \tag{7-1}$$

where $\boldsymbol{D}(\theta)$ is the "steering matrix" which maps the set of signal waveforms, $\boldsymbol{s}(t)$ onto the array and $\boldsymbol{n}(t)$ is additive noise. If the noise structure is known, the resulting covariance matrix,

$$\boldsymbol{R} = \mathrm{E}(\boldsymbol{x}\boldsymbol{x}^{\dagger}) = \boldsymbol{D}\mathrm{E}(\boldsymbol{s}\boldsymbol{s}^{\dagger})\boldsymbol{D}^{\dagger} + \mathrm{E}(\boldsymbol{n}\boldsymbol{n}^{\dagger}) = \boldsymbol{D}\boldsymbol{R}_s\boldsymbol{D} + \Sigma_n, \tag{7-2}$$

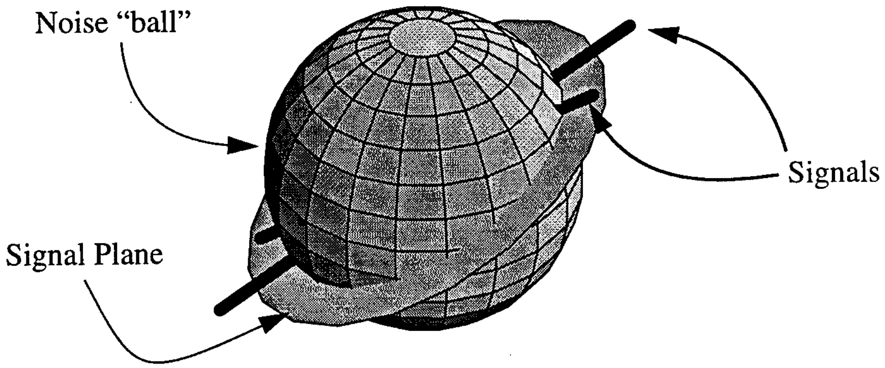


Figure 7-1: Illustration of Signal Hyperplane and Additive Noise "Ball"

may be whitened to convert Eqn (7-2) into the form,

$$\tilde{\boldsymbol{R}} = \Sigma_n^{-1/2} \boldsymbol{R} \Sigma_n^{-1/2\dagger} = \Sigma_n^{-1/2} \boldsymbol{D} \boldsymbol{R}_s \boldsymbol{D} \Sigma_n^{-1/2\dagger} + \sigma_n^2 \boldsymbol{I} = \tilde{\boldsymbol{D}} \boldsymbol{R}_s \tilde{\boldsymbol{D}}^\dagger + \sigma_n^2 \boldsymbol{I}. \qquad (7\text{-}3)$$

$\tilde{\boldsymbol{R}}$ has the structure illustrated in Figure 7-1—a noise "ball" (hypersphere) spanning the array space upon which "spikes" are inserted where the length of the spike is a function of the signal power and the orientation a function of the signal direction (or, equivalently, frequency in the event of tapped delay line rather than spatial array). The hyperplane containing the "spikes" is the signal subspace. The information-theoretic enumeration approaches (see Chapter 4) analyze the eigenvalues of $\tilde{\boldsymbol{R}}$ under assumptions of *i.i.d.* Gaussian noise to determine the multiplicity of the smallest eigenvalue; this corresponds to the dimension of the orthogonal subspace. Unfortunately, in the event of imperfect knowledge of the noise covariance structure or non-identical sensor elements, the representation of Eqn (7-2) will not be achieved and the "noise ball" will be misshapen. As a result, the information-theoretic approaches will tend to overestimate the number of arriving wavefronts. In such a case, increasing averaging spans will *not* serve to improve the signal enumeration; in fact, the opposite will occur since the information-theoretic criterion will become more confident in the uniqueness of the small eigenvalues. As a result, the information-theoretic assumptions are rarely valid for applications such as sonar signal processing; this forces a reliance on *ad hoc* enumeration techniques and detection thresholds which are subject to engineering judgement and debate.

To some extent, the inapplicability of the classical techniques is due to decoupling the enumeration from knowledge of the array manifold. Burkhardt [59] recognized that continuously transmitting sources would produce a temporally stable signal subspace. Conversely, while the orthogonal subspace would be stable, the eigenvectors spanning the orthogonal subspace would not have a preferred orientation under the classical (perfection) assumptions due to sampling effects. The validity of this observation is proved by Stewart [60] under the additive Gaussian additive noise assumption.

As will be shown in the subsequent sections, even in the event of structured noise—which causes the orientation of the noise eigenvectors to be stationary—the signal subspace stability may be exploited for effective identification and characterization of impinging wavefronts.

7.2.1 Assumptions

The methods of this chapter are not a panacea; as with previous approaches there are implicit assumptions and constraints for their application. Thus, while the subspace stability methods are valid for some situations which cannot be handled by the conventional subspace processing, they *do not* span the entire set of scenarios which are the domain of subspace processing. The restrictions and assumptions may be summarized as in Table 7-1.

Table 7-1: Subspace Stability Methods' Assumptions

Technique	*Restrictions & Assumptions*
All Subspace Stability Methods	• Continuously transmitting sources • Slowly varying source-array geometries
Burkhardt's Method	• Arbitrary array topology • Benefits from noise whitening
root-Tracker	• Restricted topology (Uniform Linear Array) • Noise whitening invalidates topology • Accurate array calibration is required
SSET	• Restricted topology (Uniform Linear Array) • Noise whitening invalidates topology

For the single-snapshot subspace processing, increasing the size of the data set results in increased accuracy of sampled covariance matrix; if the processing assumptions hold, more accurate signal enumeration and characterization result. However, if the assumptions do not hold, the misplaced confidence in the *assumed* covariance model results in enumeration errors and corresponding characterization errors. Conversely, the subspace stability methods view additional data as increasing the number of data segments—the *ensemble* producing a more accurate enumeration and characterization.

7.3 Burkhardt's Method

In his work with acoustic signals, Burkhardt [59] recognized that while MUSIC provided excellent DOA estimates when provided with an accurate signal enumeration, the characteristics of the propagation environment, additive noise, and receive array invalidated enumeration derived from information-theoretic techniques. The perturbing effects include:

structured noise	• flow noise and near-field sources • structure coupling
distributed sources	• wavefront scattering during propagation • changing source-array geometries

To defer the signal enumeration, Burkhardt proposed averaging the MUSIC spatial response derived from the assumption that given a K-element array, no more than $K-1$ signals are present. Under the assumption of $K-1$ possible signals, the MUSIC corresponds to Pisarenko's method; thus, Burkhardt's method may be viewed as the average spatial spectrum derived from sequential application of Pisarenko's method. This processing may also be viewed as analogous to the frequency spectral averaging used to suppress spurious peaks in FFT-based spectral analysis.

Defining $\boldsymbol{R}_n$ to be the covariance matrix derived from the n^{th} sampling interval and $\boldsymbol{b}_n$ to be the smallest eigenvector associated with $\boldsymbol{R}_n$, the spatial spectrum may be computed (see page 59) as

$$P_n(\theta) = \frac{1}{\boldsymbol{d}^{\dagger}(\theta)\boldsymbol{b}_n\boldsymbol{b}_n^{\dagger}\,\boldsymbol{d}(\theta)} = \frac{1}{\left\|\boldsymbol{b}_n^{\dagger}\,\boldsymbol{d}(\theta)\right\|}, \tag{7-4}$$

where $\boldsymbol{d}(\theta)$ is the steering vector mapping a hypothesized signal arriving from the DOA, θ, onto the elements of the array. Peaks in the spatial spectrum correspond to the smallest eigenvector being orthogonal to the hypothesized DOA. (The set of all possible steering vectors defines the *array manifold.*) Unfortunately, while the signal subspace constrains the smallest eigenvector to be orthogonal to the steering vector corresponding to an arriving wavefront, it does not preclude it also being orthogonal to other portions of the array manifold—which causes spurious peaks in the spatial spectrum response. This behavior is illustrated in Figure 5-3 for a single snapshot and in Figure 7-2 for a series of simulated data snapshots. In Figure 7-2 note the persistence of the peaks associated with the true signals whereas the spurious peaks occur randomly.

Averaging the spatial spectra may be accomplished either arithmetically,

$$\tilde{P}_B = \sum_n P_n \tag{7-5}$$

or geometrically,

$$P_B = \prod_n P_n . \tag{7-6}$$

The geometric average result is illustrated in Figure 7-3 and was achieved by summing the log of the response of each of the data segments. The geometric approach is attractive since it favors peak persistence rather than the strength of a individual peak. Because the peaks actually correspond to nulls in the projections rather than physically corresponding to a signal strength as would be the case with frequency spectral averaging, it may be possible for a spurious peak to appear to be very strong. Hence, we would generally prefer the geometric formulation.

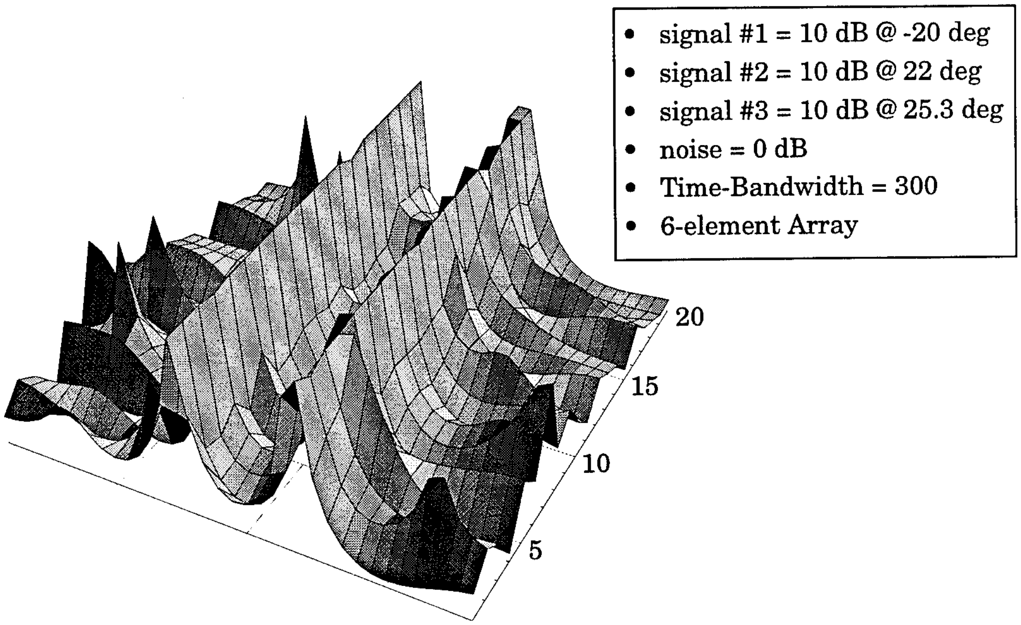


Figure 7-2: Sequential Spatial Spectrum Response of Pisarenko's Method

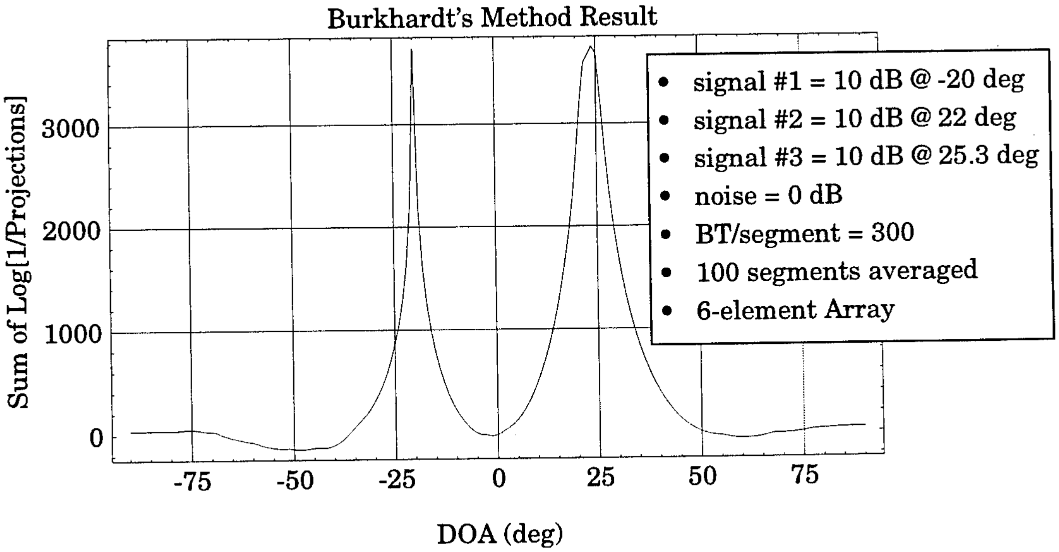


Figure 7-3: Burkhardt Method Spatial Spectrum Response

As illustrated in Figure 7-3, although separated signal sources are easily distinguished, Burkhardt's method tends to "merge" adjacent signal sources. To some extent, this is also a characteristic of all of the search-based DOA estimators (e.g., MUSIC and Min-Norm). Burkhardt proposed detecting multiple sources via analysis of the width of the spectral response. Although such an approach does not permit signal enumeration, the mere knowledge of multiple adjacent sources can be useful information.

The problems of signal enumeration are illustrated in Figure 7-4 for acoustic data collected from two signal sources by a mobile array. Under the classical assumptions, two eigenvalues should be "large" with the remaining eigenvalues approximately equal; since this is not true, the AIC and MDL criteria perceive that all eigenvalues are unique and, therefore, five signals are arriving at this six-element array. Given an accurate signal enumeration, the subspace methods can yield accurate DOA estimates—as is illustrated in Figure 7-5 using the GEESE algorithm and the assumption of two signal sources. The Pisarenko response for a sequence of covariance matrices is illustrated in Figure 7-6 and the associated Burkhardt Method spectrum in Figure 7-7. Note that the two sources are clearly visible.

Although promising, Burkhardt's method has several disadvantages:

- the enumeration of closely-spaced sources is difficult. Due to their similarity, distinguishing between multiple adjacent sources or a moving source also remains difficult;

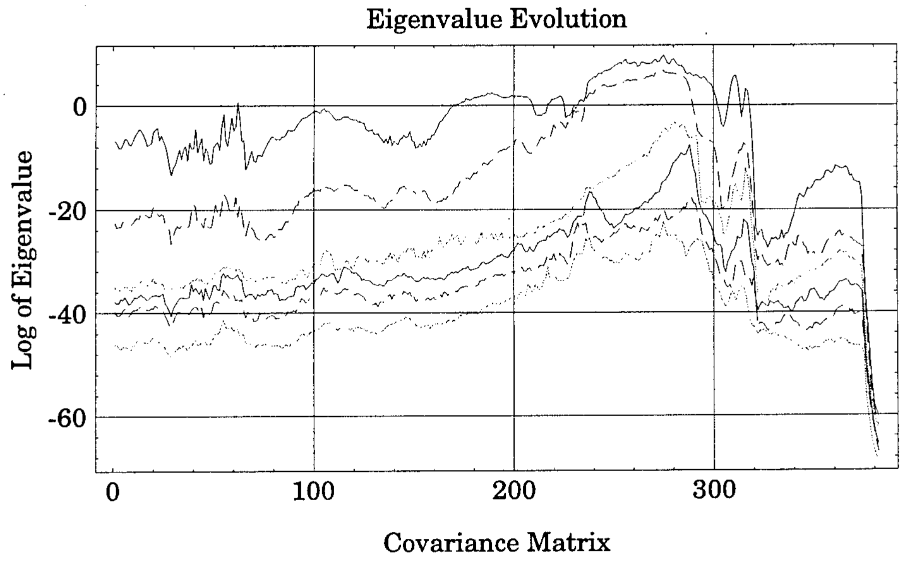


Figure 7-4: Acoustic Data Eigenvalue Trajectories

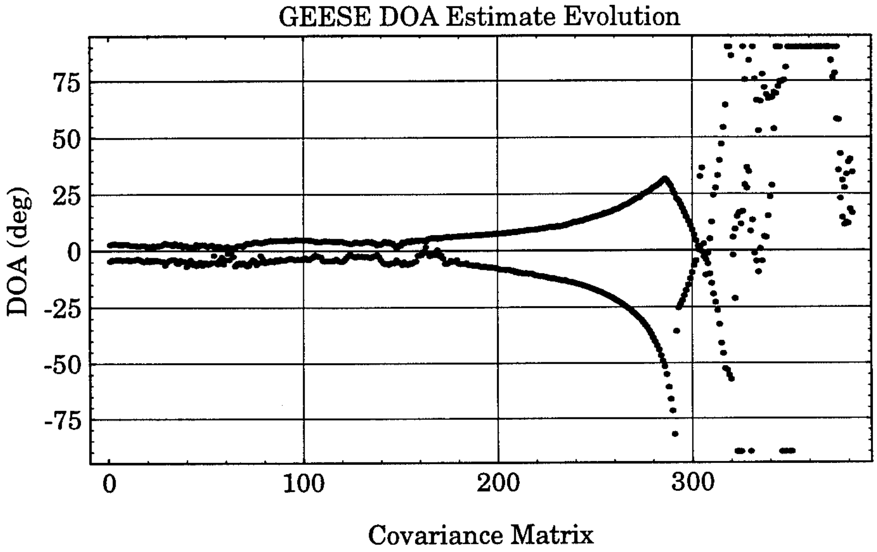


Figure 7-5: DOA Estimates Derived from Acoustic Data Assuming Two Sources

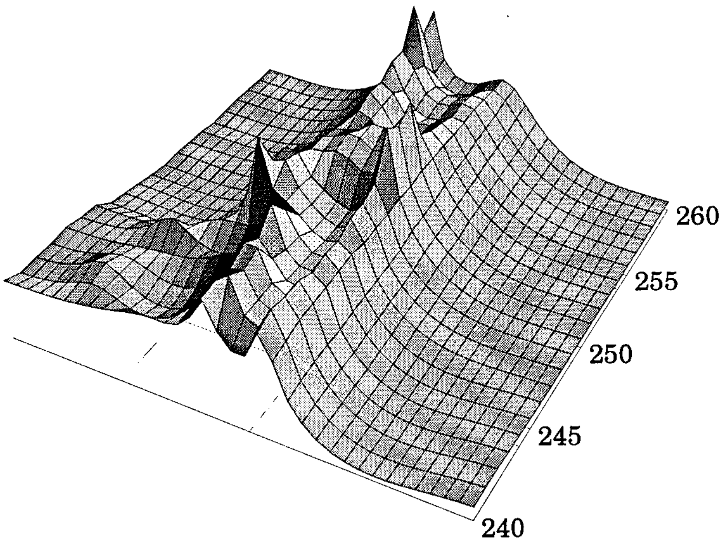


Figure 7-6: Pisarenko Response for Acoustic Data Segments {240,260}

- the computational load can be intensive since the array response must be calculated for each of the hypothesized DOAs; however, this may not be a significant constraint given the computational capacity and available RAM of modern DSPs; and
- *ad hoc* detection criteria continue to be required for realistic operational environments since attempts to identify adjacent sources via peak width requires a definition of a "wide" peak relative to a "narrow" peak.

The other approaches discussed in this chapter are conceptual children of Burkhardt's method and seek to mitigate its implementation and operational restrictions while continuing to exploit the fundamental concept of subspace stability.

7.4 Multiple-Target Tracking Concepts

The root-Tracker and SSET algorithms are defined and explored in Section 7.5 and Section 7.6, respectively. These algorithms exploit the signal subspace stability by applying *multiple target tracking* (MTT) and *multiple hypothesis tracking* (MHT) concepts to the enumeration and characterization of signal sources impinging upon a sensor array. In this section we briefly explore the physical analogue to our processing in an attempt to provide insight and intuition into the proposed subspace processing algorithms.[4] As will be shown, the melding of these aerospace-based tracking concepts with classical subspace signal processing results in robust, yet practical, high-resolution array signal processing.

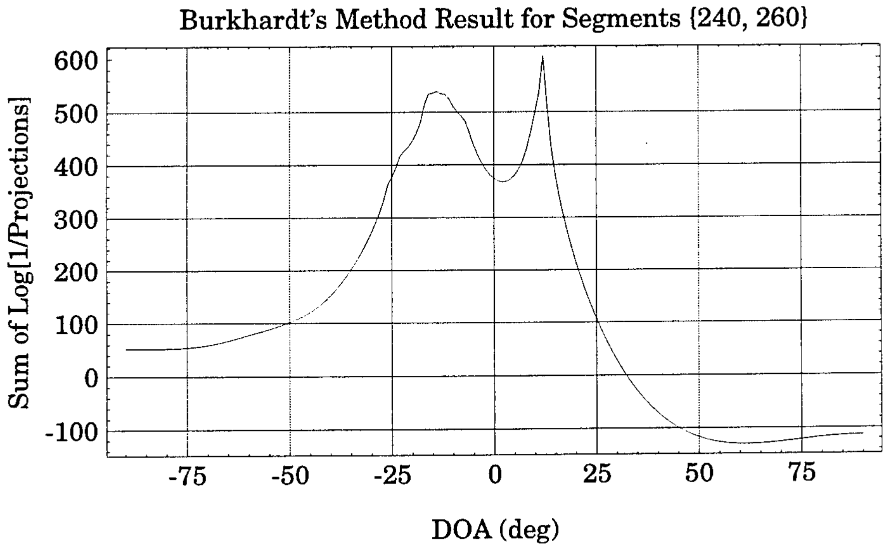


Figure 7-7: Burkhardt's Method Spatial Spectrum from Acoustic Data Segments

Multiple target tracking has its roots in radar signal processing wherein it attempted to map *events* (i.e., "detections" derived from sensor measurements and subsequent signal processing) to physical entities—i.e., planes, ships, missiles, etc. Thus, the MTT techniques were motivated by a need to assess:

- number of targets (enumeration),
- target type (characterization),
- target trajectory (history and trends)
- target intent, and
- target maneuvers.

Classically, a track will initially be identified as "tentative" with the status upgraded to "confirmed" if subsequent data validates the initial detection. Of course, quality indicators must be maintained so that unsupported hypothesized tracks may be deleted. The event-to-entity mapping—which is commonly termed "data association"—is complicated by a variety of factors which include:

- spurious detections (false alarms),
- missed detections (dropouts),
- multiple detections from a single target,
- "crossing" trajectories (indistinguishable targets), and
- maneuvering targets.

To permit a computationally feasible data association, detections from a given radar pulse are commonly *gated* so that only tracks which may be "reasonably" associated with the detection are considered in the data association. Since some detections may potentially be associated with multiple tracks, a variety of association criteria have been developed for the association. These include the nearest-neighbor filter,[5] track-splitting filter, probabilistic data association filter (PDAF), joint probabilistic data filter (JPDF), etc.

[4] Although a continuing area of research, the books by Blackman [62] and Bar-Shalom and Fortmann [63] provide a review of these concepts beyond that permissible in the relatively little space available here.

Given the difficulty of the data association problem, it was natural that *multiple hypothesis tracking* be explored. In essence, this involves adopting a probabilistic rather than deterministic perspective of the data association—deferring a definitive data association until subsequent data has corroborated that assignment; at some point, hypothesized tracks which have not been corroborated and, therefore, have a low probability must be *pruned* from the hypothesis tree. Obviously, MHT can explode into a combinatorial nightmare unless the algorithm is carefully constructed.

Fortunately, the nature of the root-Tracker and SSET problem facilitates relatively simple MTT and MHT processing. Under the classical assumptions of a calibrated array, i.i.d. additive Gaussian sensor noise, and narrowband signals, the features which are exploited are:

- signal roots will reside on the unit circle and be stationary between sampling intervals.
- spurious roots will be randomly distributed both temporally and on the complex-plane,

We are implicitly assuming constant source-array geometries. Slowly moving sources may be handled by introducing "forgetting" filters which weight recent data more heavily. Alternately, the angular velocity of the root along the unit circle may be included as one of the tracked parameters.

7.5 Root-Tracker Method

Assume a subspace method (e.g., ESPRIT, root-MUSIC, GEESE, min-Norm, etc.) is applied to a sequence of sampled covariance matrices to produce a sequence of complex-valued root sets. Furthermore, assume the root sets are generated under a *constant* hypothesis of M signals where M is the maximum number of anticiptated signals, i.e., $M \geq J$. The root-Tracker has the responsibility to assess the root set sequence and identify root trajectories having signal characteristics.

[5] For the nearest-neighbor data association, the minimum Mahalanobis distance detection-track pair is identified and the track and detection removed from further association consideration as the potential associations (determined by the gating) are analyzed for association. This process is continued until no valid pairs remain. It appears that the relatively simple nearest-neighbor filter is sufficient for the root-Tracker and SSET algorithms; however, more sophisticated association algorithms would probably be preferable if the additional complexity is not an issue.

The assumption of a constant hypothesized number of signals distinguishes the root-Tracker from the SSET algorithm discussed in Section 7.6 which uses an *adaptive* signal hypothesis approach at a price of greater algorithmic complexity and computational load. In the ensuing development, we will presume the use of Pisarenko's method to generate the root sets for consistency with Burkhardt's method. However, other algorithms are also valid and may be preferable.

As discussed in Section 5.4, if the K-element sensor array is linear with uniform sensor spacing Pisarenko's method may be implemented as a root-finding method which is less computationally demanding than the search procedure used by Burkhardt's method. Unfortunately, such an approach results in a $(K-1)^{\text{th}}$-order polynomial with J roots mapping into the DOAs of the impinging signals and $K-J-1$ spurious roots. To distinguish between true and spurious roots, Pisarenko proposed selecting the roots closest to the unit circle; unfortunately, such a strategy can fail since—although they are randomly distributed—the spurious roots have a maximum-likelihood location on the unit circle. As a result, Scharf [66] noted that "*...the technique does not work very well in practice*" [for snapshot processing].

Given our assumption of continuously transmitting and slowly moving sources, we can exploit the signal subspace stability and note that the complex-valued roots associated with signals will reside on the unit-circle and remain stable. Conversely, given i.i.d. additive sensor noise the spurious roots will be randomly located on the complex-plane with the locations independent between data segments. This is illustrated in Figure 7-8 and Figure 7-8 for the same simulated scenario as Burkhardt's method's Figure 7-2 and Figure 7-3. The problem now is to determine the root association from segment to segment. Although pattern recognition or image processing could be applied, the "tracks" illustrated in Figure 7-8 and Figure 7-8 also suggest a multiple target tracking (MTT) approach be adopted for on-line data processing. The goal is to track targets (roots) from data segment-to-segment. Confirmation of the track validity is based upon the persistence and stability of the root. Ideally, this processing should be expressible in a recursive algorithm to facilitate real-time, on-line signal detection and characterization. A very simple algorithm satisfying these objectives is proposed in the following section.

Pisarenko's method is *very* sensitive to the accuracy of the array manifold—i.e., the sensor elements must be accurately placed and the sensor amplitude and phase gains matched. Sensor gain errors will result in signal roots being perturbed on the complex-plane—resulting in erroneous DOA estimates. This behavior is illustrated in Figure 8-11 for an in-water data segment before and after array calibration. This sensitivity may be viewed as a result of fitting a high-order polynomial to the available data. Conversely, the GEESE algorithm may be

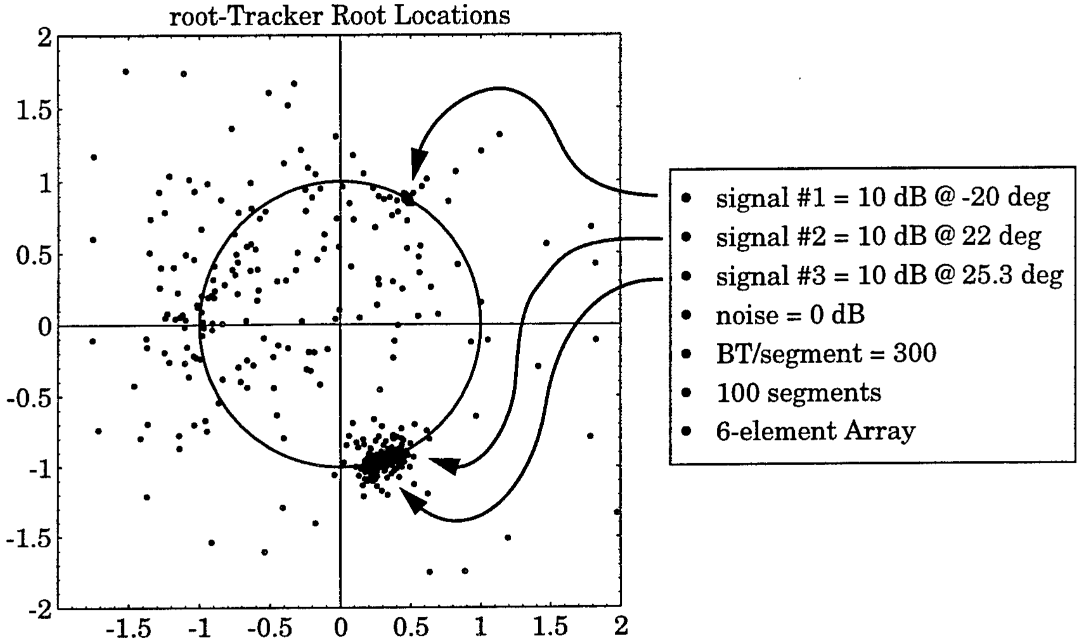


Figure 7-8: Illustration of Pisarenko Complex-Plane Root Stability

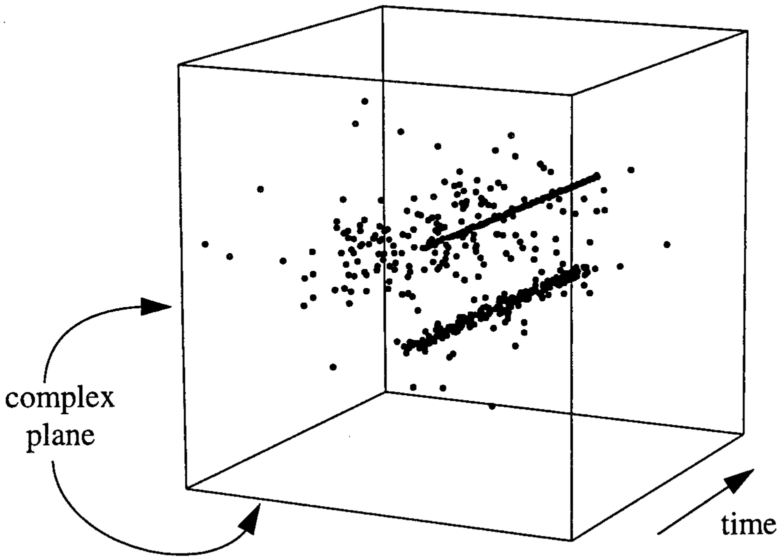


Figure 7-9: Temporal View of Pisarenko Roots

viewed as fitting a lower-order model to the sampled data—which results in more accurate and stable DOA estimates and SSET having more accurate and robust performance.

7.5.1 A Simple Root-Tracker Algorithm

The processing structure of the recursive root-Tracker implementation is illustrated in Figure 7-10. For each data segment,[6] n, the smallest eigenvector,[7] $\boldsymbol{b}_n$, of the sampled $K \times K$ covariance matrix, $\boldsymbol{R}_n$, is found. The root-finding implementation of Pisarenko's method is applied to the smallest eigenvector and the resulting $K-1$ roots, $\boldsymbol{z}_n$, processed by the root-Tracker with confirmed and tentative roots stored in a track list and confirmed tracks and their DOAs, $\boldsymbol{T}_n$, output for further processing. As alluded to in Section 7.4, there are a plethora of possible variations for a multiple-target tracker; one of the simplest possible is presented in this section. The driving assumptions in this algorithm are:

- sources are potentially moving—albeit slowly;
- nearest-neighbor data association is sufficient;
- the track mean and standard deviation should be based upon the most recent N_O observations—this finite observation span minimizes the inclusion of invalid observations in the event of moving sources;

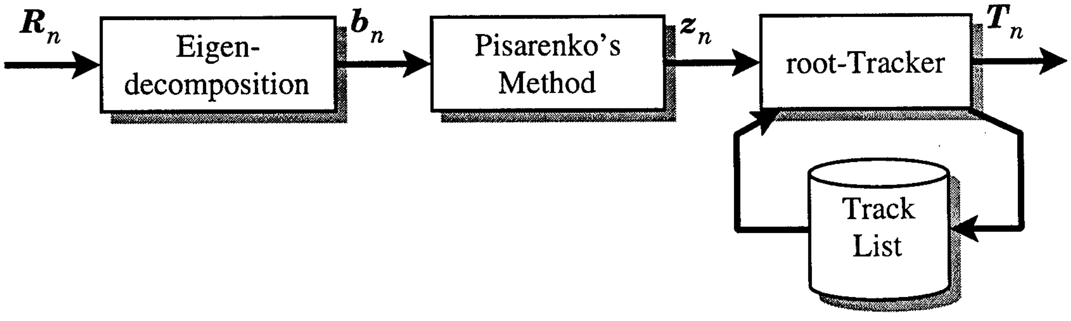


Figure 7-10: On-Line Root-Tracker Functional Architecture

[6] Under ideal noise assumptions, Stewart [60] showed that the smallest eigenvector will be very unstable. Thus, in principle we should be able to use overlapping data segments in the subspace stability processing and increase the effective size and quality of the data length. However, since the root-Tracker algorithm is predicated upon independent bases (smallest eigenvector), in practice we should opt for the safety of non-overlapping segments.

[7] Recall that the smallest eigenvector is defined as the eigenvector associated with the smallest eigenvalue. It is, of course of unit norm.

- track gate size, G_t, should adjust to reflect the stability of the roots contained in the track, $G_t \geq G_{\min}$;
- the initial gate size, G_1, should reflect the array size (root density) and operational SNRs to minimize the false alarm rate;
- track promotion from tentative to confirmed shall be based upon associated roots occurring in N_P consecutive segments; and
- confirmed track deletion shall occur if no root associations occur in N_D consecutive segments.

The *Mathematica* functions implementing these rules which were used for the root-Tracker processing in this chapter are listed in Appendix B. The implementation uses the associated roots (up to a maximum of the N_O most recent observations) to calculate the sample mean and standard deviation[8] of the sample set. The track association gate size is track-specific and ideally will correspond to the 3σ point of the track observations. To prevent the gate size errors due to insufficient observations, the gate size is adjusted via a first-order averaging filter,

$$G_n = \alpha G_{n-1} + (1-\alpha)(3\sigma_n), \qquad G_n \geq G_{\min}; \tag{7-7}$$

where,

$$\alpha = e^{-1/T}, \tag{7-8}$$

T is the time constant of the filter in units of sampling epochs, and σ_n, is the sample standard deviation estimated from the available observations. Although the gate will eventually converge to the 3σ radius, the first-order filter effectively heavily weights the initial gate size, G_1, to preclude premature gate size convergence.

During processing the most recent associated roots are maintained with the track as well as the observation mean, standard deviation, gate size, state, and the number of consecutive observations or lack thereof. Although very simple, track validation or deletion based upon consecutive hits or misses has been widely adopted in MTT systems. As discussed by Blackman [62], if the detection statistics are known, a Markov-chain analysis will yield the number of

[8] The variance of a list of N complex numbers, $\boldsymbol{z}$, which has a complex-valued mean of $\bar{z}$ is defined as

$$\sigma_{\boldsymbol{z}}^2 = \frac{1}{N-1}\|\boldsymbol{z}-\bar{z}\|^2.$$

consecutive detections, N_P, required for track promotion or number of consecutive misses, N_D, for track deletion which yield a specified false track or false deletion rates. In a similar fashion, the other operational parameters (T, N_O, G_1, and $G_{\min}$) must reflect the physics of the array environment and the anticipated array-source dynamics. Thus—consistent with physical entity MTT—operational parameters must reflect the anticipated operating environment.

7.5.2 Root-Tracker Performance

The confirmed tracks identified by the root-Tracker from the simulated data set of Figure 7-8 and Figure 7-8 is illustrated in Figure 7-11. The operational parameters were $T = 20$, $N_O = 20$, $N_P = 3$, $N_D = 2$, $G_1 = 0.2$, and $G_{\min} = 0.05$; unless otherwise noted, these values were used for all of the processing used in this section. Note the root-Tracker delay in confirming the two adjacent sources; a more sophisticated data association algorithm than nearest-neighbor would have improved the source detection performance. Although the root-Tracker works well with simulated data from this relatively small (six element) array, increasing the number of array elements to 16 sensors results in the root sequence illustrated in Figure 7-12. As might be expected, the very simple root-Tracker algorithm generates false tracks—as illustrated in Figure 7-13. However, judicious choices for initial gate sizes and track promotion criteria can reduce or eliminate the number of false tracks—as is illustrated in Figure 7-14 which shows confirmed tracks resulting from the same data set using a lengthened

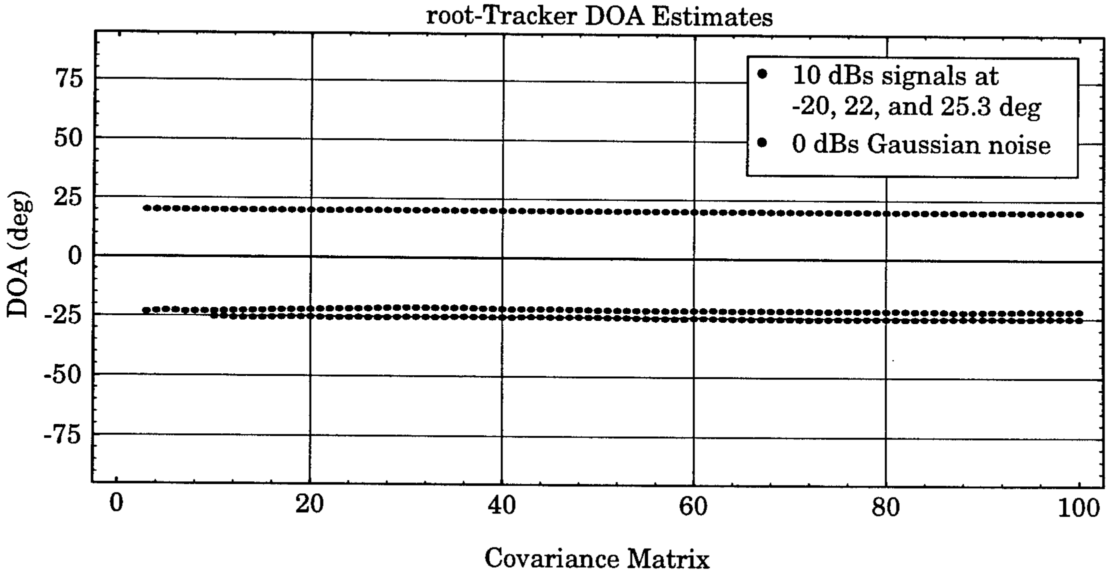


Figure 7-11: Root-Tracker Algorithm Performance

track promotion criteria (from three to four consecutive detections) and reduced initial gate size (from 0.2 to 0.1). Of course, the penalty associated with reducing the false alarm rate is an increase probability of missed detections.

Due to the high-order polynomial root finding implicit in Pisarenko's method, the root-Tracker *requires* an accurate estimate of the array manifold for accurate DOA estimates; thus, arrays must be calibrated. (See Section 8.4 for a discussion of calibration issues and exploitable features.). Additionally, the root-Tracker implicitly exploits the *instability* of the noise subspace to prevent false tracks; unfortunately, in many practical operational environments, both of these requirements are often violated. To illustrate this point, consider the root cloud

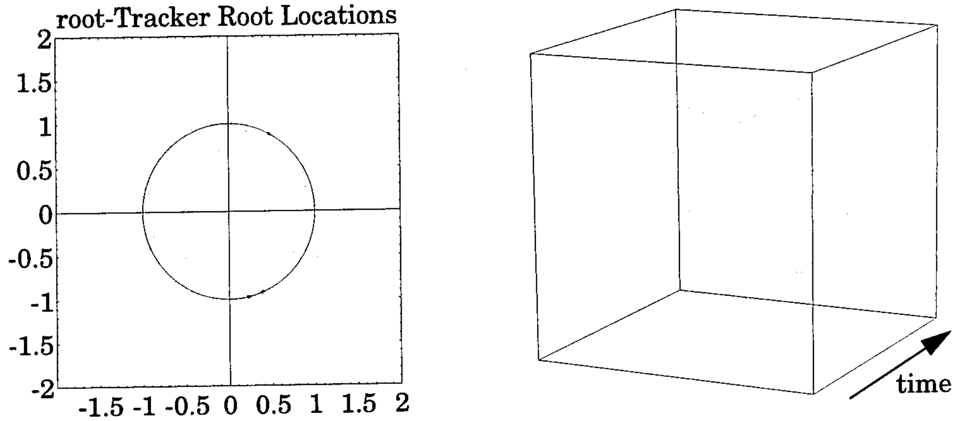


Figure 7-12: Simulated Pisarenko Root Sequence—16-Element Array

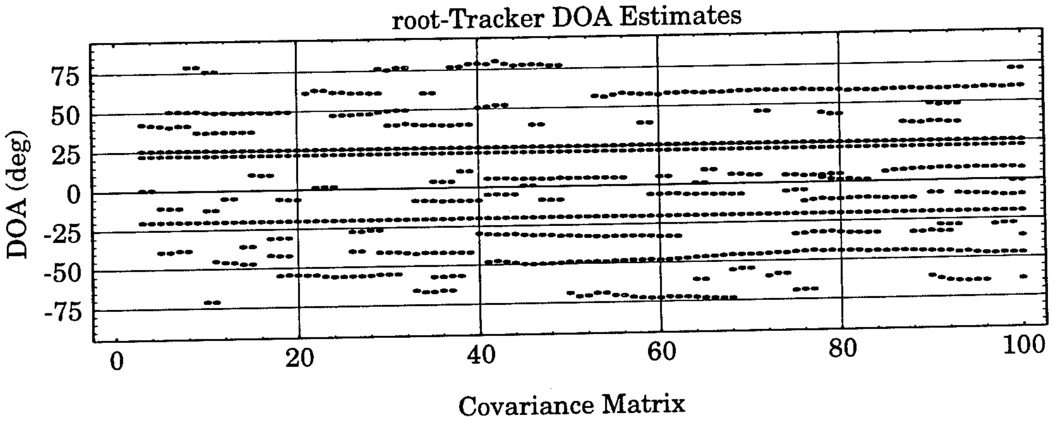


Figure 7-13: Root-Tracker Performance with 16-Element Array

evolution shown in Figure 7-15 resulting from a mobile array and two stationary sources. Here the presence of structured array noise as well as wavefront scattering during propagation result in the spurious (noise) roots being clustered. In principle, this should not pose a problem since signal roots should reside near the unit circle and, thereby, be distinguishable from the noise-induced roots which would *generally* be distributed away from the unit circle. This behavior is illustrated in Figure 7-15 for an actual data set featuring two sources. The resulting confirmed root-Tracker trajectories within ±0.2 of the unit circle are shown in Figure 7-16. Contrast this result with that obtained via the GEESE algorithm and our *a priori* knowledge of two signal sources which is also shown in Figure 7-16. While the root-Tracker

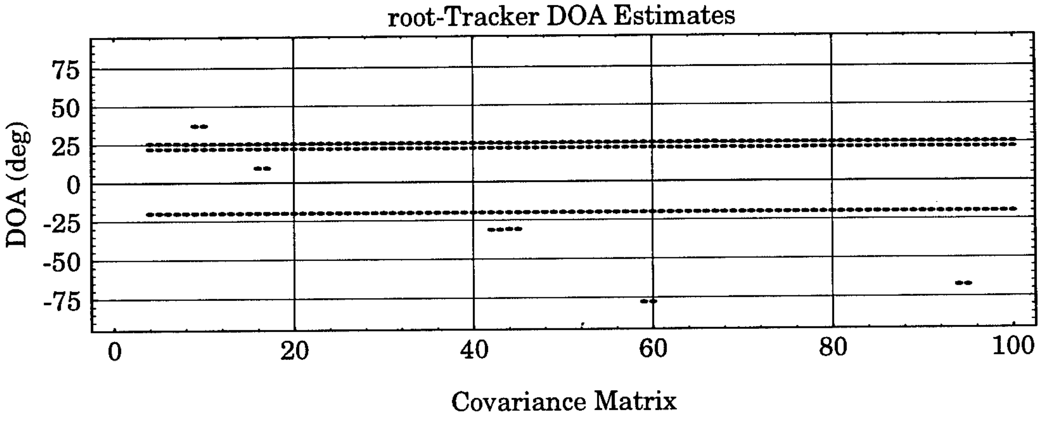


Figure 7-14: Root-Tracker Performance with Adjusted Tracking Parameters

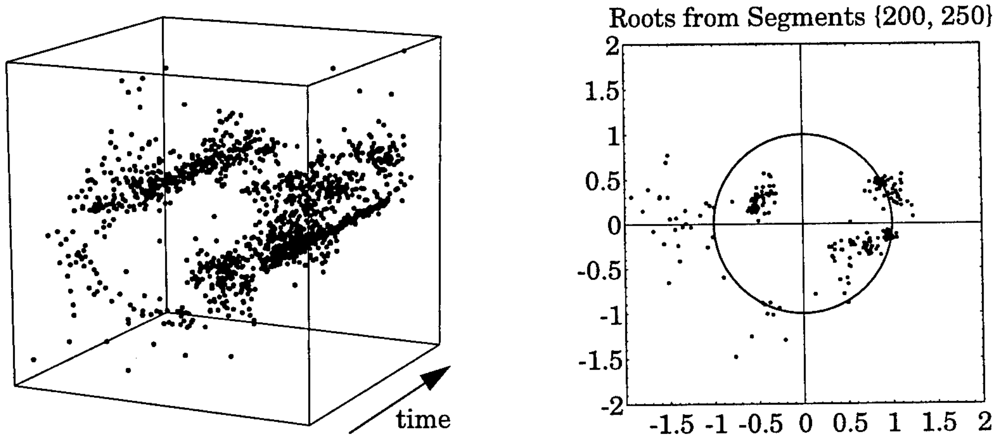


Figure 7-15: Acoustic Data Root Trajectories

accurately enumerates the signals, the DOA accuracy is marginal and the processing relies upon both the array calibration accuracy and appropriate selection of processing thresholds and track selection criteria.

7.5.3 Root-Tracker Summary

The root-Tracker merits and demerits are summarized inTable 7-2. Although the root-Tracker does not produce exceptionally accurate DOA estimates from the in-water data sets, it should be stressed that, in general, the signal *enumeration* performance exceeds that of the information-theoretic criteria and that these enumerations could be used in more classical subspace algorithms to achieve higher DOA accuracies. Additionally, as witnessed by the performance against simulated data, given an accurate array manifold and appropriate processing parameters, the root-Tracker can generate accurate DOA estimates.

Although the root-Tracker algorithm proposed in this section implements a recursive algorithm with a goal of on-line processing, the root evolutions illustrated in Figure 7-8 and Figure 7-12 reveal that the root-Tracker is only one possible algorithm of a wide class of processing approaches which

- exploit the stability of the signal subspace roots, and
- exploit the *instability* of the orthogonal subspace roots.

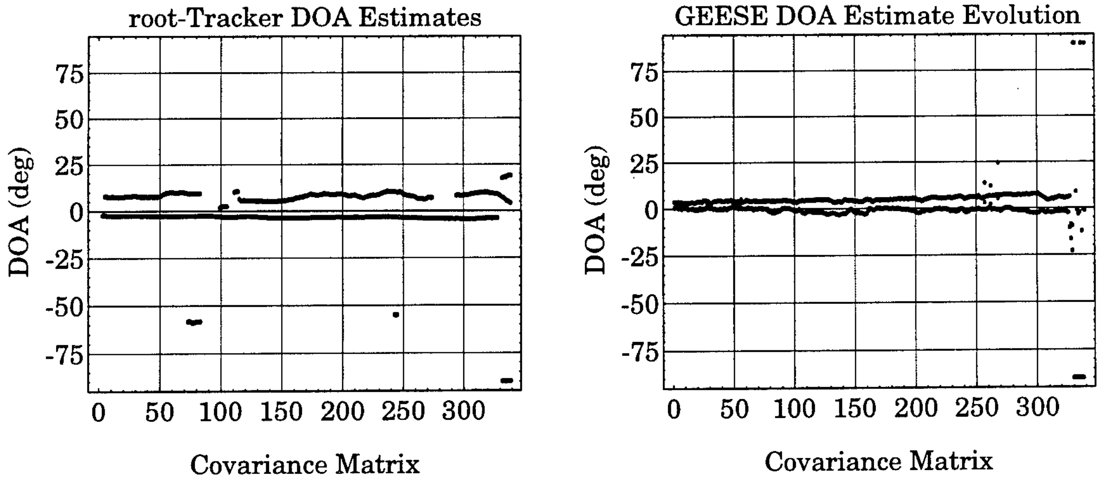


Figure 7-16: Data Set Root-Tracker and GEESE Trajectories

Table 7-2: Root-Tracker Advantages and Disadvantages

Advantages	• conceptually simple • effective given *i.i.d.* sensor noise and accurate array manifold • simple track initialization
Disadvantages	• requires an accurate array manifold • assumes a stable signal subspace *and* an unstable noise subspace • computationally demanding due to requiring roots of normally large-order polynomial • increasing array size causes computation problems as well as increasing potential for false alarms and invalidating the narrowband signal model

Natural implementations of these concepts involve applying pattern recognition and image processing techniques such as the Hough transform. However, given our interest in on-line processing techniques, we now turn our attentions to a slightly different paradigm which avoids the reliance on the noise subspace instability and, thereby, achieves improved performance against real-world data. This approach—the SSET—also has computational advantages relative to the root-Tracker and is the topic of the following section.

7.6 SSET: Subspace Stability Exploitation Tracker

Although promising, the root-Tracker has some fundamental problems due to its use of the orthogonal subspace Pisarenko's method:

- the AR signal model is easily violated by sensor errors and non-cisoid signals;
- the computational demands and model sensitivity increase with increasing array size; and,
- the assumed instability of the noise manifold may not be achieved due to structured (albeit unknown) noise.

In contrast to Pisarenko's method, the signal subspace algorithms offer computational efficiencies and DOA accuracies as well as being more robust to array manifold perturbations—*providing* an accurate signal enumeration is available. As a result, we seek an approach which exploits the signal subspace algorithms without relying upon the dubious accuracy of the information-theoretic enumeration criteria when applied to realistic data or relying on *ad hoc* techniques. The approach which we present in this section continues the root-Tracker paradigm but relaxes the restriction on the instability of the noise subspace. The **Subspace Stability Exploitation Tracker** (SSET) exploits the stability of the signal subspace and the array topology and assumes

- the signal subspace associated with J impinging wavefronts is contained within the J largest eigenvectors of the sampled covariance matrix, and
- the signal subspace DOA algorithm requirements are satisfied.

Thus, the SSET is indifferent to the noise covariance structure—providing the signals are of sufficient strength to ensure the covariance partitioning into a signal subspace and noise subspace using the sampled covariance eigendecomposition is feasible. The cost of relaxing this restriction is an *algorithmically* more complicated processing—although the computational efficiencies of the signal subspace algorithms generally result in lower *computational* demands than the root-Tracker approach.

Whereas the root-Tracker implicitly hypothesizes $K-1$ signals for each of the sequence of sampled covariance matrices, the SSET hypothesizes $\hat{J}$ wavefronts and seeks the hypothesis which maximizes a cost function which considers the location of the roots resulting from applying the GEESE algorithm to the $\hat{J}$ largest eigenvectors of the sampled covariance matrix considering both the local and global viability of the hypothesis.

To illustrate the viability of such an approach, consider the GEESE trajectories derived from a sequence of in-water covariance matrices shown in Figure 7-17 for a single signal hypothesis, Figure 7-17 for (the correct) two hypothesized signals, and in Figure 7-17 under the three signal hypothesis. (The shift at the 90^{the} covariance matrix is due to a heading change in the mobile array.) In all three figures, the complex-plane root locations are plotted for the 100–200^{th} covariance matrices. Note that under the correct hypothesis, the roots are very stable and consistent across the covariance sequence and reside on the unit circle whereas spurious roots *tend* to be distributed away from the unit circle if the number of signals is overestimated. Because of the computational efficiency of the GEESE algorithm as well as its robustness and demonstrated effectiveness, the sequel will presume GEESE as the applicable subspace processing algorithm.

The characteristics of the GEESE roots which we will exploit are:

- roots associated with signals will reside on the unit circle and be temporally stable;
- if the number of signals is over-estimated, the roots will be randomly located and *tend* to be offset from the unit-circle; and
- if the number of signals is under-estimated, the roots will reside on the unit-circle.

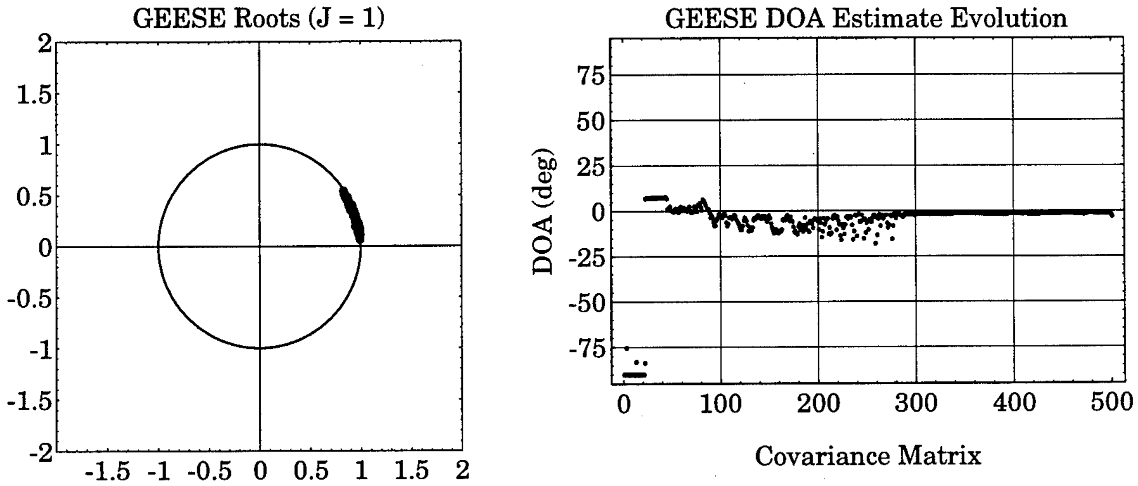


Figure 7-17: One Signal Hypothesis Root Locations and DOA Estimates

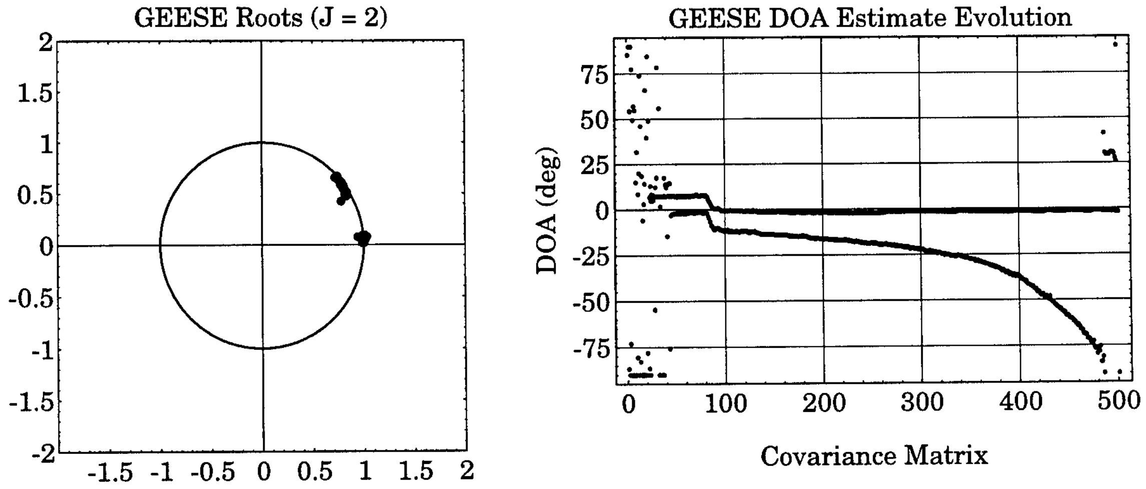


Figure 7-18: Two Signal Hypothesis Root Locations and DOA Estimate

As with the root-Tracker a variety of implementations are possible which exploit these characteristics—e.g., algorithms featuring pattern recognition and image processing techniques. However, given our implicit goal of on-line sequential processing, we again pursue a recursive implementation based upon multiple target tracking concepts.

Thus, for each covariance matrix in the sequence, the SSET processing guiding principles in estimating the number of signals from that data segment are

- maximize the number of roots on the unit circle, and
- maximize the root location consistency across covariances.

In the following section, we present a relatively simple SSET algorithm implementing these principles which is effective against both simulated and available in-water data sets. However, it should be noted that a variety of algorithms (i.e., optimization cost functions) are possible with the choice dependent upon *engineering* judgement trading off computational considerations, anticipated signal environment, and operational demands.

7.6.1 SSET Implementation Architectures and Issues

A plethora of MHT and MTT algorithms are possible to exploit the signal subspace stability; in principle, we desire the simplest algorithm which will accurately process the data from the anticipated operational environment. The generic structure of a SSET implementation is shown in Figure 7-20 and the processing components briefly described in Table 7-3.

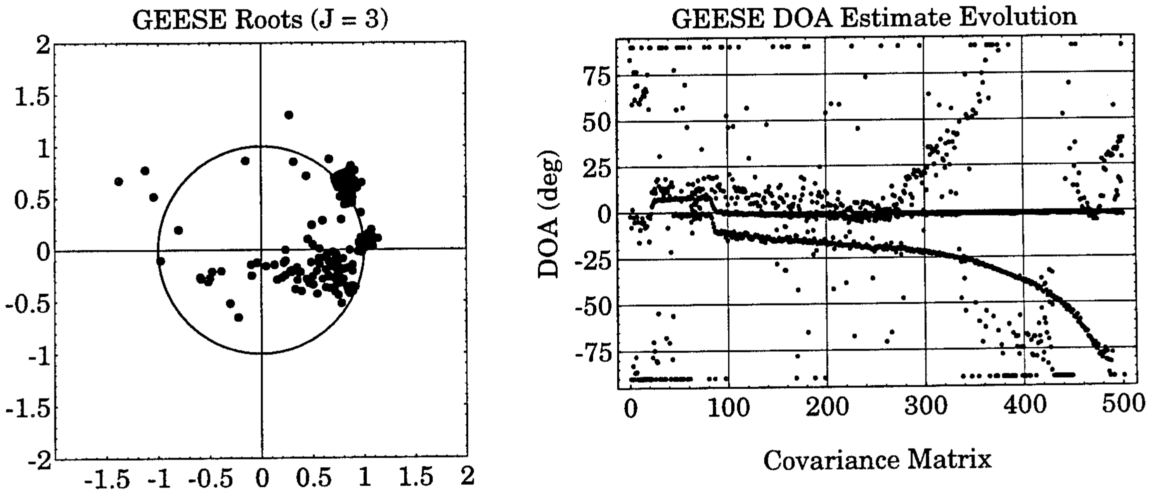


Figure 7-19: Three Signal Hypothesis Root Locations and DOA Estimate

7.6.2 A Simple SSET Algorithm

In this section we identify a relatively simple SSET algorithm which has been proven to be effective against actual as well as simulated data. Mathematica implementations of these algorithms are presented and discussed in Appendix B. In the following paragraphs we discuss some of the implementation issues associated with each of the processing components and the algorithmic approach chosen to demonstrate the effectiveness of the SSET concept against the in-water data.

Sample Covariance Matrix Generation and Eigendecomposition

The sampling interval used to estimate the array covariance matrix can be longer for the SSET algorithms than the root-Tracker since, unlike the root-Tracker which requires noise subspace *instability* as well as signal subspace *stability* for track confirmation, the SSET approach only requires signal subspace stability. The upper limit for the sampling interval must reflect the anticipated geometry dynamics as well as the model used in the track state estimation—i.e., the sampling interval must be consistent with the subspace processing assumption of *distinct* point-sources These issues are discussed in more detail in Chapter 8. In practice, the segment averaging span should be as long as possible to provide an accurate signal subspace partitioning and reduce the computational load.

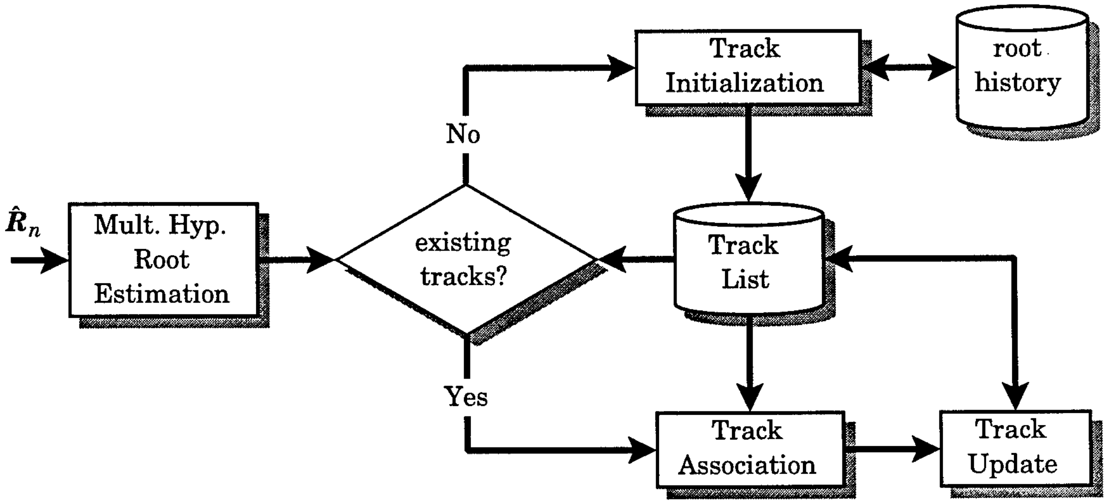


Figure 7-20: SSET Processing Flow

Table 7-3: SSET Processing Components

Multiple Hypothesis Root Estimation	• Calculates the complex-valued roots derived from a sampled covariance matrix, $\hat{\boldsymbol{R}}_n$, under various numbers of hypothesized signals. • Only those hypotheses which produce viable roots (e.g., near the unit-circle) should be output for further processing.
Track Initialization	• Establishes the baseline signal subspace and the "preferred" root locations in the track update processing. • Processing attempts to find consistent root locations across the sequence of sampled covariance matrices and identify the number and location of impinging wavefronts.
Track Association	• Given existing tracks and a new set of roots, make an "optimal" assignment of the roots to the tracks. This includes selecting the optimal hypothesized number of signals if required. • The association criterion must reflect the anticipated operating environment. The resulting algorithmic complexity can range widely.
Track Update	• Given the identified root-to-track data associations, update the existing tracks and create/delete tracks as appropriate. • The estimated track state may be a constant (for static or near-static geometries) or involve higher dimensions (e.g., location and velocity) if more dynamic environments are anticipated.
Covariance Matrix Estimation	• The interval used to generate the sampled covariance matrix is limited at the upper end by the anticipated dynamics as well as the need to establish confirmed tracks. • The lower limit is controlled by the desire to increase the effective SNR—thereby establishing the signal subspace.

If number of array elements, K, is much larger than the anticipated maximum number of signals, $J_{\max}$, then it may be computationally preferable to use a power method and compute only the signal subspace eigenvectors rather than entirely decomposing the matrix. The break-even point is in the range of $J_{\max} \approx K/4$.

Multiple Hypothesis Root Estimation

Although the hypothesis generation could be incorporated into the tracking algorithms to avoid unnecessary computations, for algorithmic simplicity the roots are calculated starting from a hypothesis $\hat{J} = 1$ and incrementing the hypothesis until the radius of a generated root deviates greater than some threshold, Δ_r, from the unit circle. Thus, a valid hypothesis satisfies,

$$||r_j| - 1| \le \Delta_r \qquad j = 1, \ldots, \hat{J}. \tag{7-9}$$

Due to array calibration and sampling errors, the hypothesized roots will not lie exactly on the unit circle. Thus, the threshold must reflect the uncertainty in the array manifold. For the examples in this chapter, a value of $\Delta_r = 0.1$ was used. This valid hypothesis criteria implicitly assumes that an AR propagation model is appropriate.

Track Initialization

Since it is possible for spurious roots (due to over-estimating the number of signals) to lie near the unit circle, a track initialization scheme is desirable which confirms the presence and stability of signals over a *sequence* of sampling intervals. The implemented algorithm requires that a common hypothesis hold for the covariance span; that is, a hypothesis $\hat{J}$, applied to each covariance matrix results in a consistent roots which are in *each* interval. Such an approach makes an implicit assumption that the number of signals remains constant during the track initialization phase.

Track initialization is required when there are no confirmed tracks; thus, the potential exists for *no* signals being confirmed. The track confirmation requirement is that the common hypothesis be valid for N_P consecutive segments with the choice of the promotion criteria, N_P, based upon anticipated dynamics and false alarm rates. In general, the threshold can be lower for the SSET approach than the root-Tracker due to the much lower false alarm rates. However, since the initialized tracks provide a reference against which subsequent data is compared, it is important that the initialized tracks accurately reflect the signal environment.

If short-term signals are anticipated, it may be desirable to implement a more sophisticated initialization algorithm wherein segment-independent hypotheses are permitted. However, this multiple-hypothesis approach increases the algorithmic complexity significantly—which is consistent with the behavior of MHT algorithms relative to MTT.

Track Association

Track association has two aspects: mapping roots from each valid hypothesized number of signals to existing tracks and, subsequently, selecting the hypothesis which provides an "optimal" association. For the mapping portion, we have adopted a simple nearest-neighbor criterion wherein the gated roots are matched to the closest available track with each root and track permitted only one association. This 1-1 mapping is consistent with the "physics" of subspace processing—the only exception being signal sources whose DOAs cross during the tracking.

The hypothesis selection aspect of track association requires a criterion to determine the "best" choice. The implemented algorithm is to:

1) associate the roots to tracks for each hypothesis,
2) determine the cumulative miss distance for each hypothesis, and
3) for those hypotheses having the maximum number of root to track associations, select the hypothesis with the minimum cumulative error.

This approach uses the established tracks as a reference to reject spurious roots due to overestimating the number of signals. Conversely, if the best fit involves more hypothesized roots than available tracks, new "probable" tracks are established.

Using the closeness-of-fit criterion implies that an accurate model of the signal source dynamics is being used. In addition to potentially allowing for source motion, changes of array orientation should be incorporated into the processing.

Track Updating

The track model used in the proposed algorithm assumes slowly-varying source-array geometries so that the DOA may be viewed as constant over the last N_O observations. If a "potential" track is validated by N_P consecutive root associations, it is promoted to "confirmed" status. Alternately, if a confirmed track has no root associations for N_D consecutive segments, it is deleted from the track list. As illustrated in Section 7.5.2, this very simple track model can effectively process in-water data.

A more sophisticated state model incorporating estimated source dynamics—e.g., angular velocity as well as the angular position (i.e., the root location on the complex plane) can yield improved DOA estimates as well as improving the track association accuracy—*providing* the more sophisticated model is appropriate. Thus, analogous to differential game theory, assuming a target behavior may result in reduced time-to-capture; however, making the minimum assumption is preferable if the target adopts optimal evasion techniques.

7.6.3 SSET Performance

The SSET performance is illustrated in Figure 7-21 for the same data set used in Figure 7-17, Figure 7-17, and Figure 7-17. This data set reflects a mobile array and two stationary signal sources. The SSET clearly identifies and accurately locates the two signal sources. Figure 7-22 illustrates the SSET performance in a scenario involving two sources plus an intermittent offset signal source transmitting during the course of the data collection. Because of the established reference signals, the SSET algorithm was able to identify and locate the intermittent signal source. To illustrate the effect of a sampling interval and model error, Figure 7-23 shows the behavior of the SSET when the mobile array initiated an uncompensated turn near the 105th covariance matrix. The maneuver causes the SSET to lose tracks; however, because the SSET algorithm used assumed stationary geometries, it was unable to establish a new track due to the effective rapid motion of the signal sources. Additionally, because of the FIR filter (five GEESE DOA roots are averaged) used for the location estimation and the presumption of

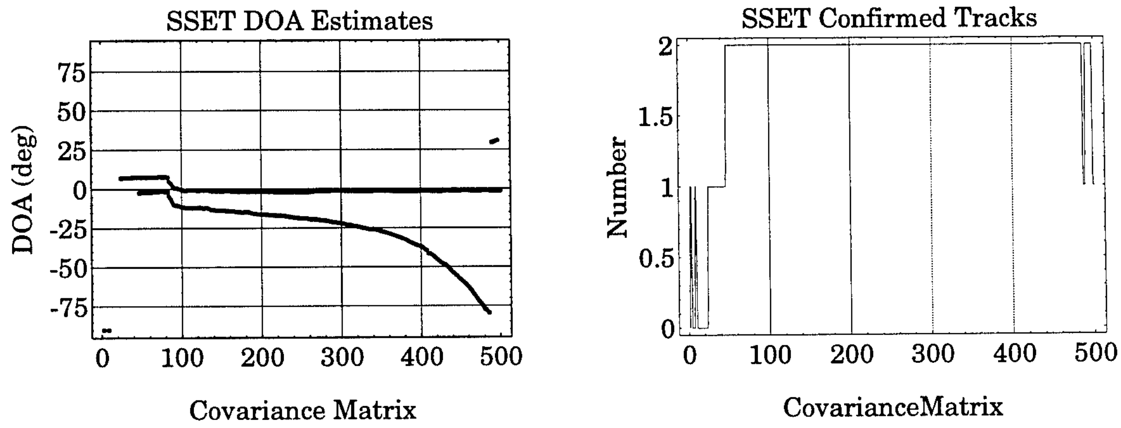


Figure 7-21: SSET DOA Estimates and Enumeration of In-Water Data

stationary geometries, there is a "group delay" in the state estimation which translates into a DOA estimation error. For reference, the GEESE trajectories under the two-signal assumption are shown in Figure 7-5. To some extent, the dynamical mismatch may be mitigated by choosing a shorter segment averaging span. This behavior is illustrated in Figure 7-24 if the averaging interval is reduced by a factor of four relative to that of Figure 7-23. As indicated, the shortened sampling interval reduces the effective dynamics of the signal sources as well as reducing the DOA estimation error introduced by the tracking filter group delay.

For the data processing, the SSET operational parameters involved an initial gate of 0.2, minimum gate of 0.05, and required valid roots to be within ±0.1 of the unit circle. Two consec-

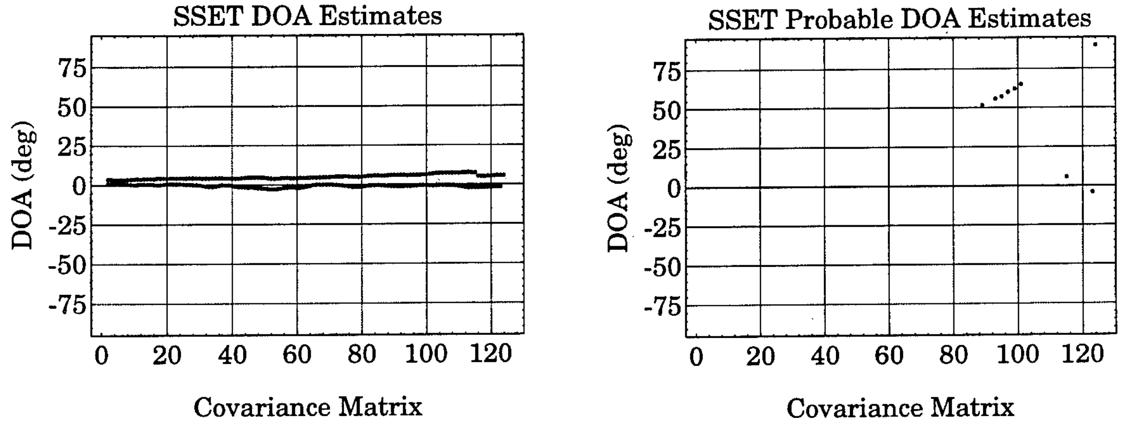


Figure 7-22: SSET Confirmed and Probable Tracks

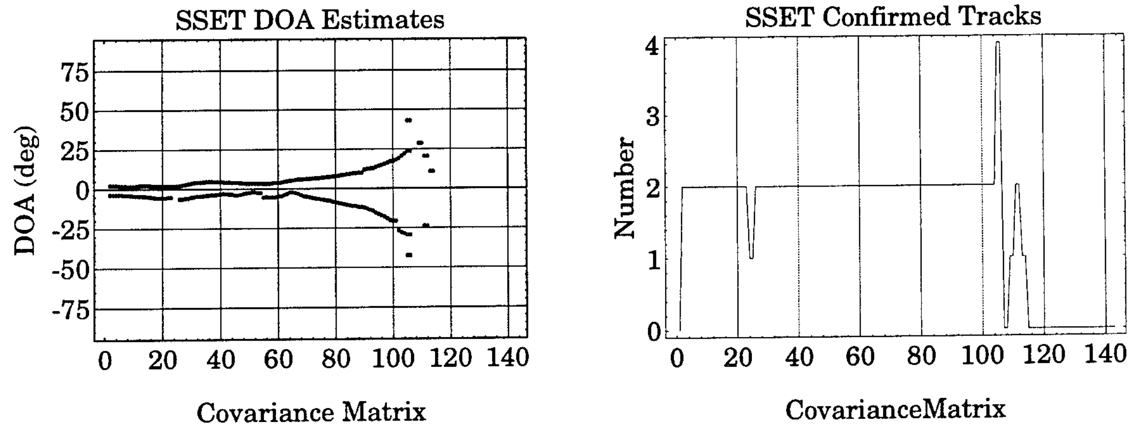


Figure 7-23: SSET Performance vs. Dynamic Geometry

utive associations were required for track promotion and two consecutive misses qualified for track deletion. The five most recent associated roots were used to estimate the signal DOA and the time constant for the gate adjustment was 20 observations.

As might be expected, the SSET also works against simulated data. The implementations of the SSET used to process the data are listed in Appendix B; note that many variations of the SSET are possible—with the choice of data association, track initialization, promotion, deletion, etc. dependent upon anticipated operating environments and the available computational resources.

7.6.4 SSET Summary

A common criticism of subspace-based algorithms has been that reduction to practice has been difficult due to the lack of an accurate yet computationally viable enumeration algorithm. To some extent, this concern has been alleviated by the SSET approach. The primary reason for the demonstrated effectiveness of SSET is the paradigm shift which couples the DOA estimation, tracking, and signal enumeration efforts without requiring multi-dimensional optimization techniques. The SSET advantages and disadvantages are summarized in Table 7-4.

As with the root-Tracker, it should be stressed that the SSET represents only one of a broad class of processing algorithms exploiting the signal subspace stability. Alternate formulations based upon image processing and pattern recognition techniques may be more appropriate for off-line processing and are topics for further research.

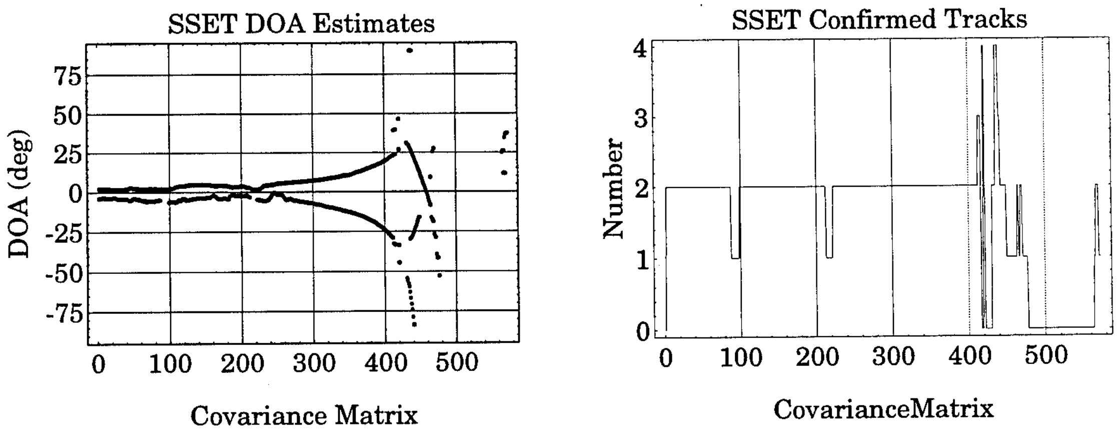


Figure 7-24: SSET Performance with Reduced Averaging Interval

Table 7-4: SSET Advantages and Disadvantages

Advantages	• effective and accurate against real data • appropriate for many physical problems • requires no assumptions concerning noise structure—other than that an eigendecomposition-based subspace partitioning is valid • relatively robust to perturbations in the assumed array manifold • real-time on-line implementation is computationally feasible
Disadvantages	• operational parameters dependent upon anticipated signal environment • restricted array topology (e.g., uniform linear)

The selection of optimal processing components for segment processing, data association, track models, etc. remains an application-specific issue with the trade-off being computational load and algorithmic complexity versus accuracy and robustness. As demonstrated, even a very simple implementation can be quite effective in an signal environment against which classical subspace processing techniques fail.

The SSET is restricted to array topologies which are amenable to the AR model implicit in signal subspace algorithms. Fortunately, this encompasses a wide number of practical arrays and applications. It is possible to apply subspace stability exploitation concepts to more general array topologies. Unfortunately, the penalty for removing the topology restriction is to introduce restrictions on the noise subspace behavior.

7.7 Conclusions and Suggestions for Further Research

Subspace-based array signal processing is predicated upon an accurate partitioning of the sampled array covariance into signal and orthogonal subspaces. In the presence of structured sensor noise or array calibration errors, signal enumeration and subsequent partitioning based upon eigenvalue analysis requires *ad hoc* criteria since information-theoretic approaches fail. The subspace stability methods introduced in this chapter address those concerns in scenarios featuring near-static source-array geometries and continuously transmitting sources. The three variations of subspace stability methods which have been examined in this chapter are presented in Table 7-5 along with some of the salient features and restrictions.

The most attractive subspace stability method appears to be the SSET approach. SSET has been demonstrated to effectively process in-water data sets against which conventional single-snapshot subspace processing fails and, as such, represents a significant advance in practical real-time implementation of subspace methods to in-water sonar array processing.

Table 7-5: Subspace Stability Exploitation Methods Summary

Burkhardt's Method	• averages spatial response of Pisarenko's method • requires signal subspace stability *and* orthogonal subspace instability • computationally demanding • arbitrary array topology • low resolution for signal enumeration
root-Tracker	• assumes a fixed number of hypothesized signals • requires signal subspace stability *and* orthogonal subspace instability • identifies signals using MTT techniques • limited to uniform linear arrays • computationally efficient • vulnerability to model errors increases with increasing array size
SSET	• combines multiple-hypothesis enumeration with MTT techniques • requires *some* signal subspace stability • limited to uniform linear arrays • computationally efficient • model error sensitivity decreases with increasing array size

Although the relatively simple SSET algorithm proposed in this chapter has been demonstrated against in-water data, further research remains with respect to track initialization, data association, and track promotion/deletion criteria. The implemented SSET algorithm used the GEESE signal subspace algorithm; other signal subspace algorithms should be evaluated with respect to their suitability and practicality for incorporation into the SSET concept.

In this chapter, we have focussed on multiple-target tracking techniques as a mechanism to exploit the signal subspace stability; other approaches which may be viable include applying pattern recognition and image processing techniques to the sequence of SSET roots.

Finally, the identification of a computationally viable subspace stability method applicable to arbitrary array topologies and robust with respect to assumptions concerning the sensor characteristics, noise environment, and scenario dynamics remains an open research topic.

8. Sensor and Model Errors

In the preceding chapters, an implicit assumption of perfect sensors, signals, and noise was made:

Narrowband Signals	The assumption of narrowband signals implies that a cisoid model is appropriate—so that a wavefront's contribution at element at any given instant in time is simply a phase-shifted version of that contributed to the other elements.
Equal Power and Sensor-Independent Noise	This implies that the additive noise at the array elements is equi-powered and independent and due to independent noise processes. This permits the noise to be modelled as a diagonal matrix, $\sigma^2 \boldsymbol{I}$.
Perfect Sensors	The array elements are assumed to have unity gain as well as sample the signal at *exactly* the same epoch—thus, neither amplitude or phase errors are introduced by the array.
Point Sources and Static Geometry	If the signal sources move during the signal averaging interval used to develop the sampled covariance matrix, the signal subspace will be "blurred"—essentially corresponding to a distributed source. This makes the signal detection problem more difficult as well as precluding using "long" sampling intervals to improve the effective signal-to-noise ratio.

These assumptions are, to some extent, required to make the signal detection and location estimation tractable. In this chapter we will briefly review the characteristics, sources, and significance of deviations from these assumptions and methods to mitigate their influence on the subspace processing accuracy.

8.1 Narrowband Signals

A central assumption of the subspace methods is that at any sampling epoch the impinging (complex-valued) wavefront at each array element may be modelled as a phase-shifted replica of the wavefront value at any other array element/sensor *at that same sampling epoch*.Viewing the wavefront waveform as a baseband band-limited signal, $s_B(t)$, translated to a higher frequency,

$$s(t) = e^{\imath 2\pi f_0 t} s_B(t), \tag{8-1}$$

the narrowband assumption implies that the nominal center frequency, f_0, is much greater than the signal bandwidth, β,

$$f_0 \gg \beta. \tag{8-2}$$

Under this assumption, the array covariance matrix given a single impinging wavefront is Hermitian and Toeplitz with a constant magnitude for *each* matrix element and constant phases along the diagonals (assuming negligible noise contributions). While this approximation is generally appropriate for radar applications, it is often violated for sonar. As the signal bandwidth increases relative to the nominal center frequency, the *magnitude* of the autocorrelation of the signal will drop off so that the sampled covariance matrix will not have constant magnitude. Although the *expected* phase shifts will still be correct, the phase standard deviation will increase as the sensor separation increases. In essence, the mapping of the signal onto the array can no longer be separated into geometric and temporal components, i.e., $\boldsymbol{x}(t) \neq \boldsymbol{d}(\theta)s(t)$.

To illustrate this point, assume that we have a band-limited analytic signal which has the spectrum,

$$S(f) = \begin{cases} \dfrac{2\pi}{\beta} & |f-f_0| \le \dfrac{\beta}{2} \\ 0 & |f-f_0| > \dfrac{\beta}{2} \end{cases}, \tag{8-3}$$

where the constant in-band amplitude of $2\pi/\beta$ is chosen to normalize the expression for the autocorrelation function so that $R(0) = 1$. This spectrum is illustrated in Figure 8-1 for a ten percent bandwidth signal; ten percent bandwidth is a commonly accepted threshold for defining a signal as being narrowband—however, as will be illustrated, the definition of narrowband in the subspace processing sense must consider the array size as well as the signal detection criteria. The autocorrelation of a signal is the Fourier transform of the power spectral density, i.e.,

$$R(\tau) = \frac{1}{2\pi}\int_{-\infty}^{\infty} S(w)e^{-\imath wt}d\omega \qquad (8\text{-}4)$$

where, by definition, $R(\tau) = -R^*(-\tau)$ since the power spectrum, $S(\omega)$, must be real-valued. Applying Eqn (8-3) to Eqn (8-4) and integrating leads to the expression,

$$R(\tau) = e^{\imath 2\pi f_0\tau}\frac{\sin\pi\beta\tau}{\pi\beta\tau} = e^{-2\pi f_0\tau}\operatorname{sinc}(\pi\beta\tau). \qquad (8\text{-}5)$$

This equation tells us that samples separated by a time $1/\beta$ will be uncorrelated—which is consistent with standard sampling theory. Furthermore, the *expected*[1] phase shift between samples will depend upon the time delay and the *band-center frequency*—which validates the concept of using the nominal frequency to calculate the steering vectors used in the array processing. Finally, note that the *amplitude* of the autocorrelation will vary as a function of the signal bandwidth as well as the time delay. It is this last effect which poses problems for the array processing.

The autocorrelation may also be expressed in terms of the fractional bandwidth, $\hat{\beta}$, relative to the carrier frequency and the spatial separation, $\hat{\tau}$, expressed in units of cycles at the nominal frequency. Thus, with,

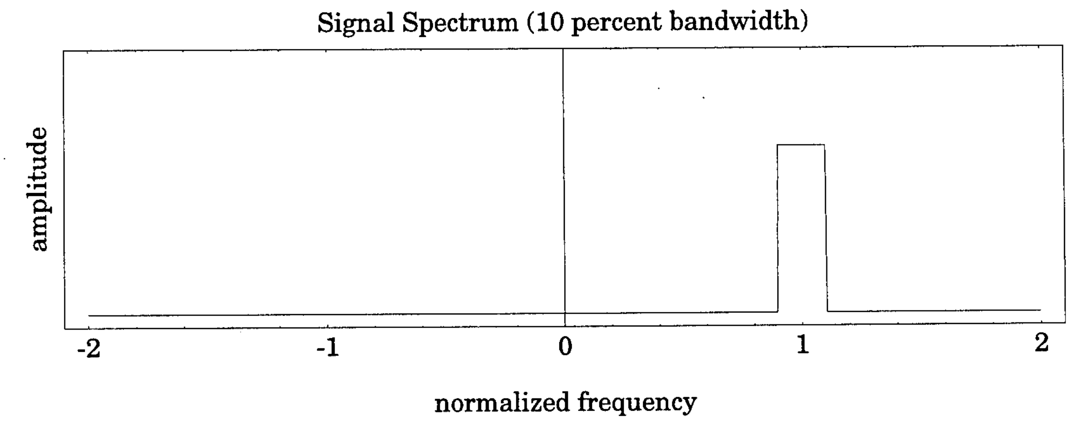


Figure 8-1: Illustration of Idealized 10 Percent Bandwidth Signal Spectrum

[1] Since the power spectral density of Eqn (8-3) corresponds to the expected spectrum of the band-limited signal, Eqn (8-5) is the expected autocorrelation behavior. The actual sampled signal spectrum will, in general, be perturbed from this idealized assumption.

$$\hat{\beta} = \frac{\beta}{f_0}, \tag{8-6}$$

and,

$$\hat{\tau} = f_0 t, \tag{8-7}$$

the autocorrelation function of Eqn (8-5) may be expressed as,

$$R(\hat{\tau}) = e^{-\iota 2\pi\hat{\tau}}\operatorname{sinc}(\pi\hat{\beta}\hat{\tau}). \tag{8-8}$$

Eqn (8-8) implies that for "large" fractional signal bandwidths the assumption that the signal vector received at the array elements, $\boldsymbol{x}(t)$, may *not* be decomposed into a geometry dependent term and a waveform dependent term, i.e.,

$$\boldsymbol{x}(t) = \boldsymbol{x}(\boldsymbol{s}(t), \theta) \neq \boldsymbol{D}(\theta)\boldsymbol{s}(t) \tag{8-9}$$

The autocorrelation amplitude behavior is illustrated in Figure 8-2 for a ten percent fractional bandwidth and as a function of both time delay and fractional bandwidth in Figure 8-3. Hence, the term "narrowband" depends upon the array aperture as well as the signal bandwidth and center frequency; with typical half-wavelength element spacings, the fundamental

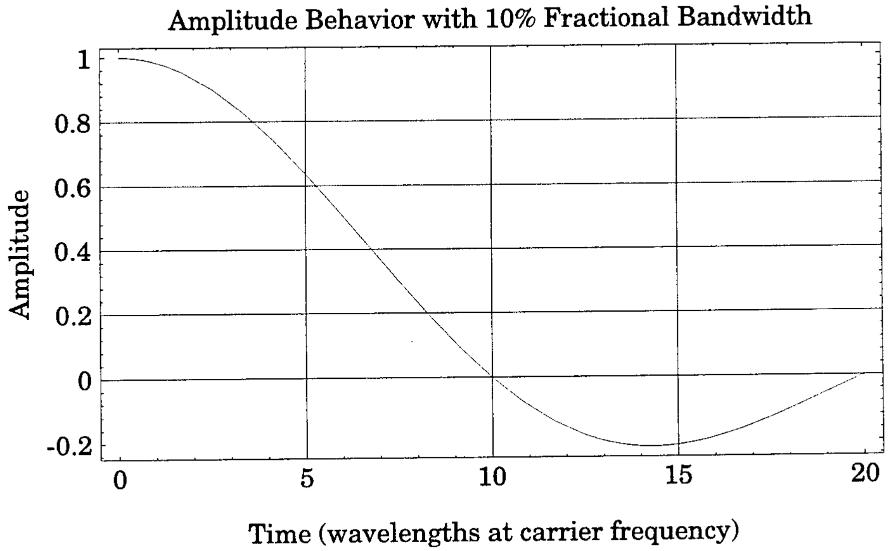


Figure 8-2: Auto-Correlation Amplitude Behavior Illustration

narrowband assumption can be violated with array sizes and fractional bandwidths encountered in acoustic signal processing.

Figure 8-4 and Figure 8-5 show the effect of increasing fractional bandwidth on the eigenvalues of the sampled covariance matrix for a single signal in the absence of noise—under the classical assumptions, there should be *one* non-zero eigenvalue. Since the signal detection algorithms typically analyze the covariance matrix eigenvalues (see Chapter 4), such behavior

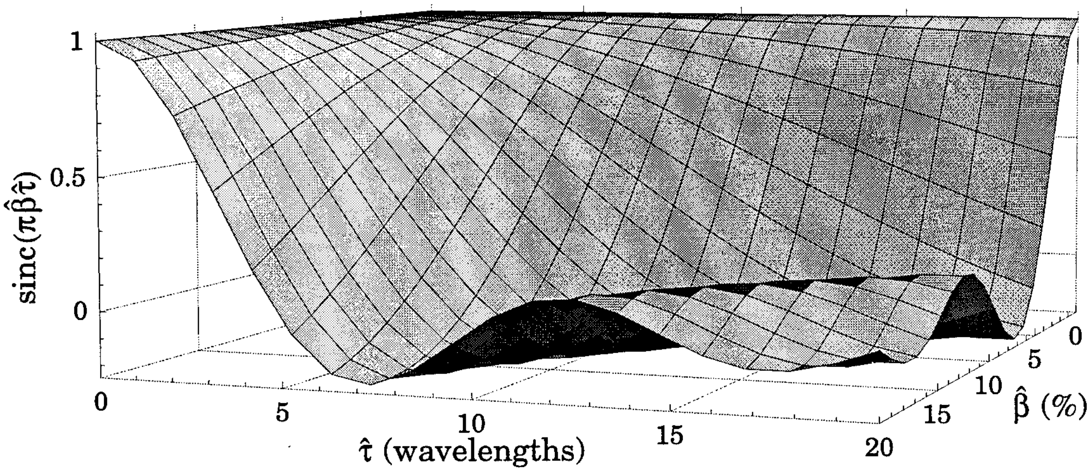


Figure 8-3: Autocorrelation Amplitude vs. Delay and Fractional Bandwidth

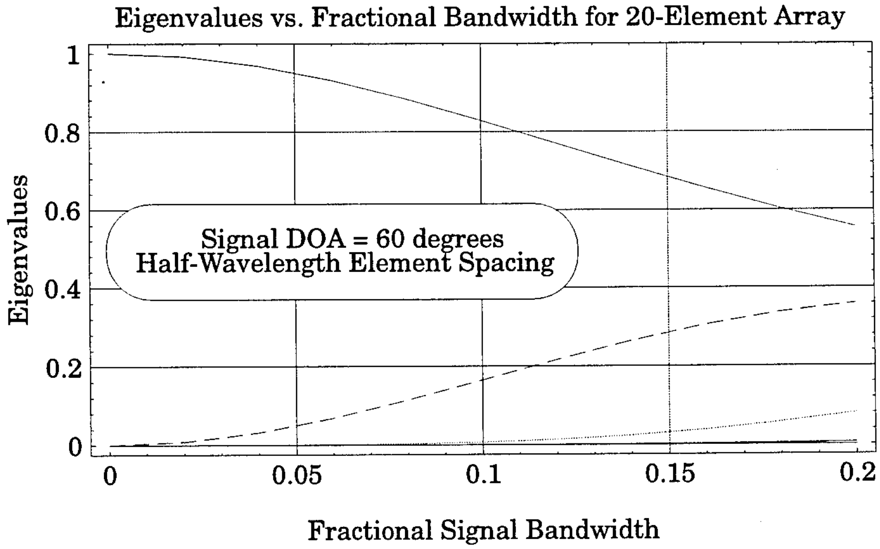


Figure 8-4: Eigenvalue Behavior vs. Fractional Bandwidth for a 20-Element Array

will, in general, result in overestimating the number of impinging signals. The eigenvalues of Figure 8-5 are plotted on a log scale in Figure 8-7; note that for fractional signal bandwidths of only five percent approximately 20 dB of eigenvalue "coupling" will be introduced by the this geometry and bandwidth-dependent effect—in general, this results in a reduced dynamic range as well as invalidating the fundamental assumptions of the information-theoretic

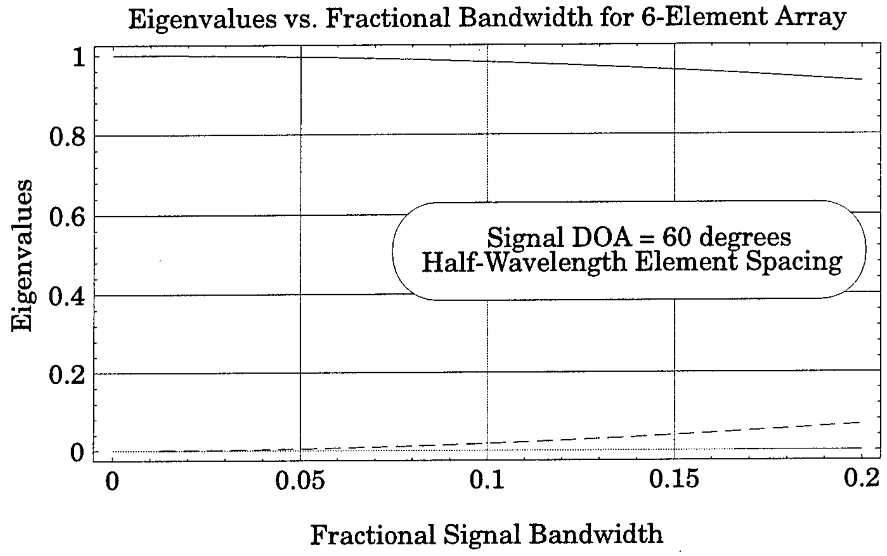


Figure 8-5: Eigenvalue Behavior as a Function of Bandwidth for a Six-Element Array

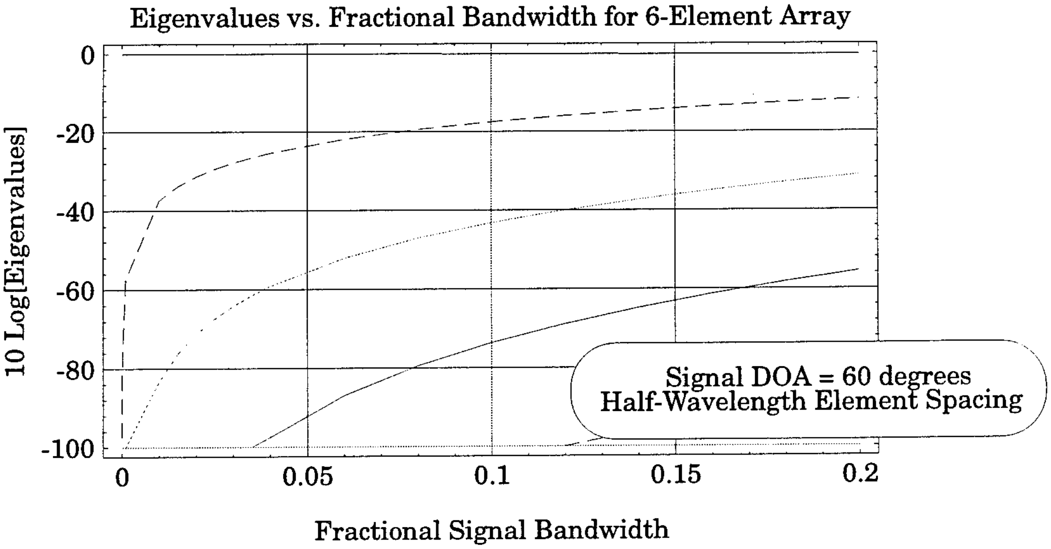


Figure 8-6: Eigenvalues vs. Fractional Bandwidth for a Six Element Array

detection criteria. Figure 8-7 illustrates the spatial response of a ten percent bandwidth signal impinging on a 6-element array with the (incorrect) MDL-based estimate of two signals.

The effect of the signal bandwidth on the sampled covariance matrix *cannot* be removed without knowledge of the source directions and fractional bandwidths. Note, however, that even with cisoid signals the magnitude of the (nominally Toeplitz structured) array covariance matrix elements will vary as a function of source geometry if multiple sources are impinging on the array. In practice, we can avoid the problems associated with a wideband signal by applying a Fourier decomposition to the signal and processing each frequency band independently—thereby generating a situation where the narrowband approximation *is* valid—albeit at a cost of greater computational load and complexity.

In summary, the narrowband signal model is essential to subspace-based array processing because it permits decoupling the geometric and temporal components of the signal, i.e., modelling the received signal as

$$\boldsymbol{x}(t) = \boldsymbol{D}(\theta)\boldsymbol{s}(t) + \boldsymbol{n}(t). \tag{8-10}$$

This, in turn, permits a subspace partitioning into a J-dimensional signal subspace and a $K-J$-dimensional orthogonal subspace. Violating the narrowband assumption results in eigenvalue coupling—which violates information-theoretic signal enumeration assumptions. Violating the narrowband assumption also perturbs the subspace partitioning so that the DOA estimation algorithms are also affected. The matrix-shifting algorithms (e.g., GEESE or

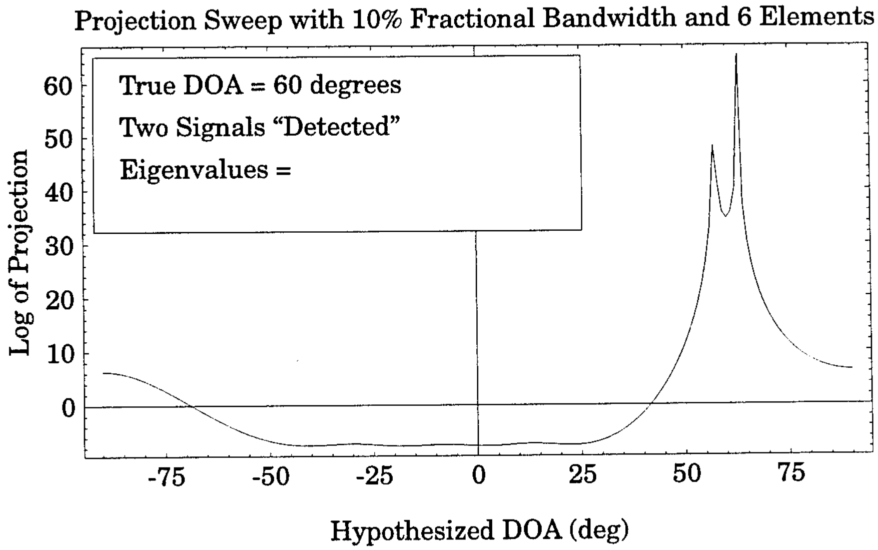
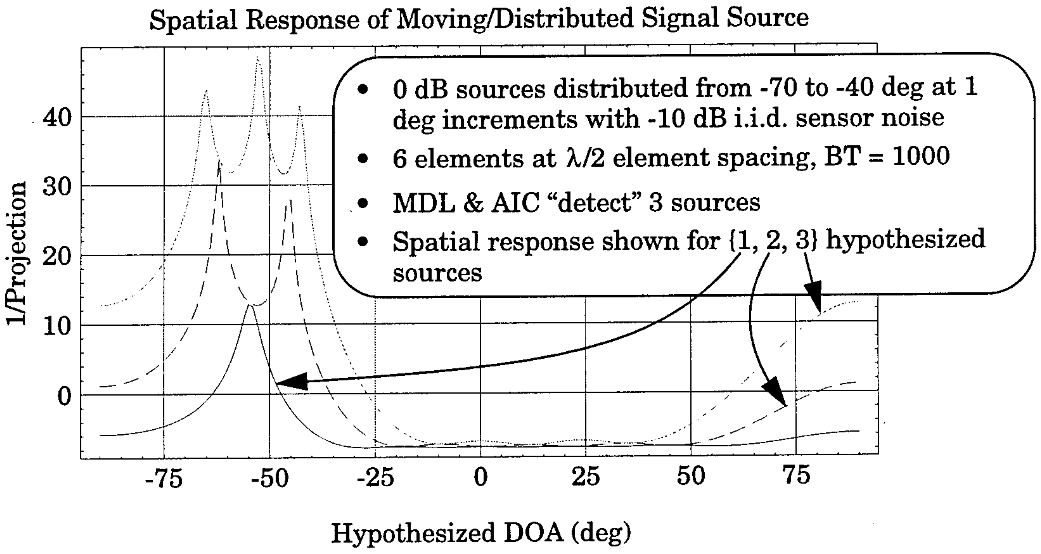


Figure 8-7: MUSIC Sweep showing Effect of Fractional Bandwidth

ESPRIT) are less vulnerable to the narrowband violation since they only require the narrowband assumption to be valid over the translational shift between subarrays. In fact, since the ESPRIT algorithm does not employ a subspace partitioning, it should be relatively resistant to narrowband assumption violations—providing an accurate estimate of the noise covariance matrix is achievable. This would appear to make ESPRIT or GEESE the preferred subspace processing algorithm for the root-Tracker algorithm presented in Section 7.5.

8.2 Distributed and Moving Sources

Just as violating the narrowband assumption will disturb the perturb the idealized structure of the sampled covariance matrix, distributed or moving sources perturb the covariance structure. Subspace methods implicitly presume a signal environment featuring stationary (in the physical sense) *point* sources.[2] If the sampled covariance matrix is derived from a moving source—or, equivalently, a distributed source—*fundamental assumptions* about the signal characteristics will be violated and result in *spurious detections* and DOA estimates. This is illustrated in Figure 8-8 which shows the spatial (MUSIC) response due to 31 sources uniformly distributed from -40 to -70 degrees impinging upon a 6-element array.


Spatial Response of Moving/Distributed Signal Source

- 0 dB sources distributed from -70 to -40 deg at 1 deg increments with -10 dB i.i.d. sensor noise
- 6 elements at λ/2 element spacing, BT = 1000
- MDL & AIC "detect" 3 sources
- Spatial response shown for {1, 2, 3} hypothesized sources

Figure 8-8: Spatial Response Derived from a Distributed Source

[2] Generally, a far-field signal source is presumed; however, MD-MUSIC permits a near-field assumption—with a commensurate increase in computational complexity and loading.

Multi-path or reflections from physical structures such as boundaries or wakes can approximate the presence of a distributed source. If the signal sources are in motion, integration intervals must be chosen such that the stationary point-source geometries is valid. Although this may not be a problem for many applications, for underwater acoustics, the combination of low data rates and high speed relative to sound speed may preclude long integration in intervals. In such cases, the subspace stability methods of Chapter 7 may be useful.

In summary, if scenario geometries are dynamic, the sampled covariance matrix integration interval must be chosen to avoid seriously violating the static geometry assumption.

8.3 Colored Noise Processing

Although many of the subspace detection and DOA algorithms may be criticized for the presumption of "perfect noise"—i.e., a diagonal covariance matrix, $\sigma^2 \boldsymbol{I}$—*some* assumptions about the noise structure are required; otherwise, *any* sampled covariance matrix could be rationalized as due to noise contributions. If the noise structure were known, it could be removed from the sampled covariance matrix; however, such information is not available for many applications.

In the event of a high SNR scenario, deviating from the "perfect noise" assumption will not affect the DOA estimates; however, since the noise eigenvalues will not be derived from a "ball" in the K-dimensional pace of the sampled covariance matrix, the assumption that all noise eigenvalues will be equal will be violated—which can confuse the information theoretic approaches to signal detection. If the noise structure were known or adequately estimated, such effects could be removed by "whitening" the sampled covariance matrix—essentially warping the subspace so that the noise is a ball. Thus, given a (full-rank) noise covariance matrix of the form, $\sigma^2\Sigma_n$, where σ^2 is an unknown real-valued scalar corresponding to the noise *level*, so that the sampled covariance matrix is,

$$\boldsymbol{R} = \boldsymbol{D}\boldsymbol{R}_s\boldsymbol{D}^\dagger + \sigma^2\Sigma_n. \tag{8-11}$$

Decomposing the (Hermitian) model of the noise covariance *structure* matrix, Σ_n, into

$$\Sigma_n = \boldsymbol{U}_n\Lambda_n\boldsymbol{U}_n^\dagger \tag{8-12}$$

leads to the representation for the inverse as,

$$\Sigma_n^{-1} = \boldsymbol{U}_n\Lambda_n^{-1}\boldsymbol{U}_n^{\dagger} = (\boldsymbol{U}_n\Lambda_n^{-1/2})\,(\Lambda_n^{-1/2}\boldsymbol{U}_n^{\dagger}) = \Sigma_n^{-1/2\dagger}\,\Sigma_n^{-1/2} = \boldsymbol{S}^{\dagger}\boldsymbol{S} \tag{8-13}$$

where we have implicitly defined,

$$\boldsymbol{S} = \Sigma_n^{-1/2} = \Lambda_n^{-1/2}\boldsymbol{U}_n^{\dagger} \tag{8-14}$$

and exploited the fact that the eigenvalue matrix, Λ_n is real-valued and diagonal due to the sampled covariance matrix being Hermitian. Applying this similarity transform to the sampled covariance matrix yields,

$$\tilde{\boldsymbol{R}} = \boldsymbol{SRS}^{\dagger} = \boldsymbol{SDR}_s\boldsymbol{D}^{\dagger}\boldsymbol{S}^{\dagger} + \sigma^2\boldsymbol{S}\Sigma_n\boldsymbol{S}^{\dagger} = \tilde{\boldsymbol{D}}\boldsymbol{R}_s\tilde{\boldsymbol{D}}^{\dagger} + \sigma^2\boldsymbol{I}. \tag{8-15}$$

After application of the similarity transform, detection algorithms based upon equal noise eigenvalues will again be valid. However, this procedure will also warp the array manifold such that

$$\tilde{\boldsymbol{d}}(\theta) = \boldsymbol{Sd}(\theta) \tag{8-16}$$

should be used in the DOA search algorithms. In general, the application of the similarity transform implies that algorithms which exploit specific array topologies—e.g., ESPRIT, GEESE, etc.—will no longer be valid. In such a case, the identified signal or noise subspace would need to be warped back to the correct array topology prior to DOA processing.

8.4 Perfect Sensors and Array Calibration

As noted by Fuhrman [47], *"high-resolution direction finding methods ... require precise knowledge of the array manifold"*—or, equivalently, they require sensors which do not introduce sensor-specific gains and phase shifts to the collected data or introduce channel coupling. Although sensor arrays are typically aligned to some extent during construction, over time the component drifts or physical perturbations of the array cause the array to deviate from the idealized behavior. Thus, it is desirable to calibrate the array in its operational environment and estimate parameters which can be used to mitigate these effects. Although a number of researchers [42]–[49] have addressed the issue of auto-calibration, these approaches impose requirements for either calibration signals from known locations, limitations on the array topology, ideal noise covariance structures, or knowledge of the number of impinging signals which either make the methods computationally demanding or impractical for on-line operations. In this section we introduce an apparently novel calibration algorithm which exploits the Toeplitz array structure resulting from plane waves impinging upon a uniform linear array. The intent is a computationally efficient algorithm which is complementary to the root-Burkhardt subspace stability exploitation method discussed in Section 7.5.

8.4.1 Assumptions and Notation

The auto-calibration approach presented in the sequence relies upon a number of assumptions:

- the array is linear with a constant sensor spacing;
- at least one signal is impinging on the array;
- the signals are narrowband;
- there is negligible cross-channel coupling;
- the additive sensor noise covariance may either be modelled as $\sigma^2 \boldsymbol{I}$ or is small relative to the impinging signals;
- the impinging wavefronts are planar and uncorrelated; and
- the signals and additive noise are uncorrelated.

The net effect of these assumptions is that the expected array covariance matrix in the absence of sensor gain and phase errors will be Toeplitz—i.e., constant along the diagonals. The signals arriving at the K elements of the array may be modelled as

$$\boldsymbol{x}(t) = \boldsymbol{D}(\theta)\boldsymbol{s}(t) + \boldsymbol{n}(t) \tag{8-17}$$

where $\boldsymbol{s}(t)$ is a vector of the signals, $\boldsymbol{D}$ is the geometry-dependent mapping of the signals onto the array elements and $\boldsymbol{n}(t)$ is the sensor additive noise (as discussed in detail in Chapter 2). If the received signals are then perturbed by a sensor-specific complex gain, the sampled signal from the array, $\boldsymbol{y}(t)$, is

$$\boldsymbol{y}(t) = \boldsymbol{G}\boldsymbol{x}(t) = \boldsymbol{G}\,[\boldsymbol{D}(\theta)\boldsymbol{s}(t) + \boldsymbol{n}(t)] \tag{8-18}$$

where $\boldsymbol{G}$ is a complex-valued diagonal gain matrix,

$$\boldsymbol{G} = \begin{bmatrix} g_0 & & & \\ & g_1 & \mathbf{0} & \\ & \mathbf{0} & \ldots & \\ & & & g_{K-1} \end{bmatrix} = \mathrm{Diag}(\boldsymbol{g}). \tag{8-19}$$

The elements of the complex-valued gain vector, $\boldsymbol{g}$, are defined as

$$g_k = \gamma_k e^{\iota\psi} \qquad k = 0, 1, \ldots, K-1 \tag{8-20}$$

where γ_k is a positive scalar representing the amplitude gain and $-\pi \le \psi < \pi$ is a scalar representing the introduced phase shift. Employing the Schur (Hadamard) product, Eqn (8-18) may be represented in the equivalent form,

$$\boldsymbol{y}(t) = \boldsymbol{g} \otimes \boldsymbol{x}(t) = \boldsymbol{g} \otimes (\boldsymbol{Ds} + \boldsymbol{n}), \tag{8-21}$$

where "$\boldsymbol{C} = \boldsymbol{A} \otimes \boldsymbol{B}$" denotes the Schur (element-by-element) multiplication such that $c_{ij} = a_{ij} b_{ij}$.

The perceived covariance matrix, $\tilde{\boldsymbol{R}}$, is related to the covariance matrix in the absence of gain perturbations, $\boldsymbol{R}$, by

$$\tilde{\boldsymbol{R}} = \boldsymbol{GRG}^\dagger = \boldsymbol{G}\mathrm{E}((\boldsymbol{Ds} + \boldsymbol{n})\,(\boldsymbol{Ds} + \boldsymbol{n})^\dagger)\boldsymbol{G}^\dagger \tag{8-22}$$

where, if we assume that the signals and noise are uncorrelated, Eqn (8-22) reduces to

$$\tilde{\boldsymbol{R}} = \boldsymbol{G}(\boldsymbol{DR}_s\boldsymbol{D}^\dagger + \Sigma_n)\boldsymbol{G}^\dagger = \boldsymbol{gg}^\dagger \otimes (\boldsymbol{DR}_s\boldsymbol{D}^\dagger + \Sigma_n). \tag{8-23}$$

Under the classical assumption of i.i.d additive noise, the noise contribution to the covariance will converge to the form $\sigma^2\boldsymbol{I}$ so that $\boldsymbol{R}$ will be Toeplitz (constant along the matrix diagonals) as well as being Hermitian. Even if the noise is spatially colored, if the impinging signals are "strong", the expected covariance matrix will be approximately Toeplitz. Under the Toeplitz assumption, the expected covariance matrix may be represented as

$$\boldsymbol{R} = \begin{bmatrix} a_0 & a_1 e^{\iota\varphi_1} & a_2 e^{\iota\varphi_2} & \ldots & a_{K-1} e^{\iota\varphi_{K-1}} \\ a_1 e^{-\iota\varphi_1} & a_0 & a_1 e^{\iota\varphi_1} & & \\ a_2 e^{-\iota\varphi_2} & a_1 e^{-\iota\varphi_1} & a_0 & & a_2 e^{\iota\varphi_2} \\ \vdots & & & & a_1 e^{\iota\varphi_1} \\ a_{K-1} e^{-\iota\varphi_{K-1}} & \ldots & a_2 e^{-\iota\varphi_2} & a_1 e^{-\iota\varphi_1} & a_0 \end{bmatrix}. \tag{8-24}$$

Recall that implicit in the representation of Eqn (8-24) is the assumption that *at least one signal is impinging* on the array and that the effects of the noise-induced perturbations is negligible.

8.4.2 Amplitude Gain Estimation

Paulraj and Kailath [42] noted that the estimation of the amplitude gains may be mapped into a linear equation representation by taking the logarithm of Eqn (8-23) under the Toeplitz assumption. Using only the upper triangle of the $K \times K$ covariance matrix (since the lower portion is redundant) we have $K(K+1)/2$ equations and $2K$ unknowns—the gains associated with the K elements, γ_k, and the K amplitudes of the diagonals, a_{k-1}, of the assumed covariance matrix which is illustrated in Eqn (8-24). To illustrate, consider the equations resulting from a three-element array,

$$\begin{bmatrix} 2 & 0 & 0 & 1 & 0 & 0 \\ 0 & 2 & 0 & 1 & 0 & 0 \\ 0 & 0 & 2 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 0 & 1 \end{bmatrix} \log \begin{bmatrix} \gamma_0 \\ \gamma_1 \\ \gamma_2 \\ a_0 \\ a_1 \\ a_2 \end{bmatrix} = \log \begin{bmatrix} |\hat{R}_{11}| \\ |\hat{R}_{22}| \\ |\hat{R}_{33}| \\ |\hat{R}_{12}| \\ |\hat{R}_{23}| \\ |\hat{R}_{13}| \end{bmatrix}, \tag{8-25}$$

where the structure of Eqn (8-25) is achieved by moving along successive diagonals of the perturbed sampled covariance matrix, $\hat{\boldsymbol{R}}$. Let us adopt the notation that

$$\Gamma = \left[\log(\gamma_0)\ \log(\gamma_1)\ \ldots\ \log(\gamma_{K-1})\right]^{\mathrm{T}}, \tag{8-26}$$

$$\boldsymbol{a} = \left[\log a_0\ \log a_1\ \ldots\ \log a_{K-1}\right]^{\mathrm{T}}, \tag{8-27}$$

and

$$\boldsymbol{r} = \log \left| \begin{bmatrix} \operatorname{diag}(\hat{\boldsymbol{R}}, 0) \\ \operatorname{diag}(\hat{\boldsymbol{R}}, 1) \\ \vdots \\ \operatorname{diag}(\hat{\boldsymbol{R}}, K-1) \end{bmatrix} \right| \tag{8-28}$$

with $\mathrm{diag}(\boldsymbol{A}, n)$ representing a vector derived from the n^{th} diagonal of the matrix, $\boldsymbol{A}$, and the understanding that the $|\cdot|$ and $\log(\cdot)$ in Eqn (8-28) apply to the *elements* of the resulting vector. Using this notation, Eqn (8-25) may be expressed in the general form,

$$\boldsymbol{B}\begin{bmatrix}\Gamma\\ \boldsymbol{a}\end{bmatrix} = \hat{\boldsymbol{r}} \tag{8-29}$$

where the structure of the coefficient matrix, $\boldsymbol{B}$, is implicitly defined by the ordering of the measurement vector, $\hat{\boldsymbol{r}}$ and the number of array elements. Note that we *cannot* use Eqn (8-29) to solve for the sensor amplitude gains and estimated diagonals via the classical least-squares use of the pseudo-inverse, i.e.,

$$\begin{bmatrix}\Gamma\\ \boldsymbol{a}\end{bmatrix} \neq (\boldsymbol{B}^{\dagger}\boldsymbol{B})^{-1}\boldsymbol{B}^{\dagger}\hat{\boldsymbol{r}}, \tag{8-30}$$

since the coefficient matrix, $\boldsymbol{B}$, is singular—thus, the gains can only be determined to within a scale factor. Paulraj and Kailath [42] proposed handling this problem by eliminating the diagonal level terms, $\boldsymbol{a}$, from Eqn (8-29); using all possible combinations of the equations produces a set of $K(K^2-1)/2$ equations in K unknowns—which is still singular. Using a pseudo-inverse defined using the $K-1$ non-zero values and vectors resulting from a SVD enables a minimum-norm solution of the gain amplitude estimation—within an arbitrary scale factor. The disadvantage of this approach is that it involves considerable manipulation of the measurement vector with a commensurate increase in the algorithmic complexity.

As an alternative, note that if we make an arbitrary assignment concerning the received power level or assign an element as the reference having a gain of "one", then the singularity is removed and the sensor gains may be estimated without resorting to a singular value decomposition. Thus, using the definition that the "true" power received at an element is the average of the main diagonal of the received covariance matrix, i.e.,

$$a_0 \equiv \hat{a}_0 = \frac{1}{K}\sum_{k=1}^{K}\hat{R}_{kk}. \tag{8-31}$$

we achieve a non-singular system of equations. Realizing that the equation involving the $\hat{R}_{1K}$ term does not contribute to the estimate of the gains, we can define a new system of equations of the form,

$$\underset{\sim}{\boldsymbol{B}}\begin{bmatrix}\Gamma\\ \underset{\sim}{\boldsymbol{a}}\end{bmatrix} = \underset{\sim}{\hat{\boldsymbol{r}}}, \tag{8-32}$$

with the definitions,

$$\underset{\sim}{\boldsymbol{a}} = \begin{bmatrix}\log a_1 \;\ldots\; \log a_{K-1}\end{bmatrix}^{\mathrm{T}}, \tag{8-33}$$

and

$$\boldsymbol{r} = \log\left|\begin{bmatrix}\operatorname{diag}(\hat{\boldsymbol{R}}, 0) - \hat{a}_0\\ \operatorname{diag}(\hat{\boldsymbol{R}}, 1)\\ \vdots\\ \operatorname{diag}(\hat{\boldsymbol{R}}, K-1)\end{bmatrix}\right| \tag{8-34}$$

and $\underset{\sim}{\boldsymbol{B}}$ is similar in form to $\boldsymbol{B}$ with the deletion of the column associated with a_0 and the elimination of the last row. Under these assumptions, the system of equations for a three-element array becomes,

$$\begin{bmatrix}2&0&0&0&0\\0&2&0&0&0\\0&0&2&0&0\\1&1&0&1&0\\0&1&1&1&0\end{bmatrix}\log\begin{bmatrix}\gamma_0\\ \gamma_1\\ \gamma_2\\ a_1\\ a_2\end{bmatrix} = \log\begin{bmatrix}|\hat{R}_{11}| - a_0\\ |\hat{R}_{22}| - a_0\\ |\hat{R}_{33}| - a_0\\ |\hat{R}_{12}|\\ |\hat{R}_{23}|\end{bmatrix}. \tag{8-35}$$

Using the pseudo-inverse,

$$\underset{\sim}{\boldsymbol{B}}^{\#} = (\underset{\sim}{\boldsymbol{B}}^{\dagger}\underset{\sim}{\boldsymbol{B}})^{-1}\underset{\sim}{\boldsymbol{B}}^{\dagger} \tag{8-36}$$

the sensor amplitude gains may be estimated as

$$\begin{bmatrix}\Gamma\\ \underset{\sim}{\boldsymbol{a}}\end{bmatrix} = \underset{\sim}{\boldsymbol{B}}^{\#}\underset{\sim}{\hat{\boldsymbol{r}}}. \tag{8-37}$$

From an implementation perspective, the pseudo-inverse would be calculated initially and reused for each gain estimation since it is a constant dependent only upon the number of sensors in the array—thus, the least-squares estimation of the amplitude gains may be accomplished via a rearrangement of the sampled covariance matrix into a vector followed by a matrix-vector multiply. Furthermore, note that if the diagonal level estimates are not if interest, the rows of $\underset{\sim}{B}^{\#}$ associated with the unused parameters may be deleted and the computational load further reduced. Figure 8-9 illustrates the effect of the amplitude gain adjustment on a covariance matrix derived from in-water data and a 6-element ceramic array.

8.4.3 Phase Gain Estimation

The estimation of the sensor phase error is similar to that used for the amplitude estimation since the phase along the covariance diagonals in the presence of strong uncorrelated signals will also be constant as illustrated in Eqn (8-24). Since only the super diagonals of the covariance matrix above the main diagonal provide non-redundant phase gain information, we have $K(K-1)/2$ equations for $2K-1$ unknowns—the phases associated with the K elements and the $K-1$ phases along each of the super diagonals. Using the definition, $\arg(z) = \operatorname{atan}(\operatorname{Re}(z)/\operatorname{Im}(z))$, permits expressing those equations as a linear system. The equations may be illustrated for a 5-element array:

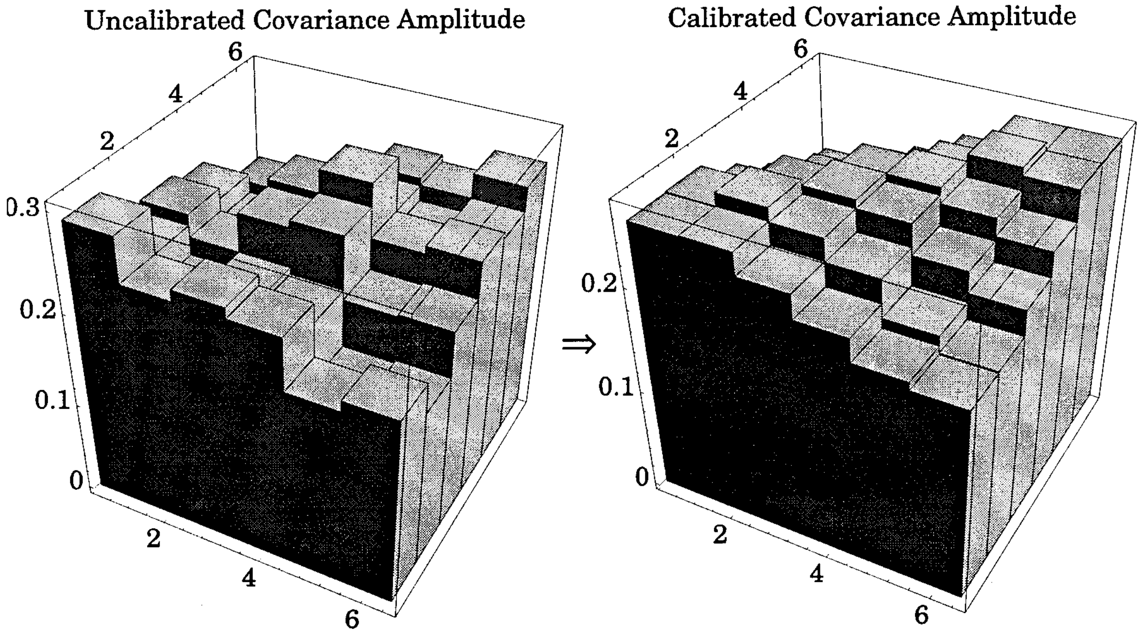


Figure 8-9: Illustration of the Effect of Amplitude Gain Adjustment

$$\begin{bmatrix} 1 & -1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & -1 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & -1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & -1 & 1 & 0 & 0 & 0 \\ 1 & 0 & -1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & -1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & -1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & -1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & -1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \psi_0 \\ \psi_1 \\ \psi_2 \\ \psi_3 \\ \psi_4 \\ \varphi_1 \\ \varphi_2 \\ \varphi_3 \\ \varphi_4 \end{bmatrix} = \arg \begin{bmatrix} \hat{R}_{12} \\ \hat{R}_{23} \\ \hat{R}_{34} \\ \hat{R}_{45} \\ \hat{R}_{13} \\ \hat{R}_{24} \\ \hat{R}_{35} \\ \hat{R}_{14} \\ \hat{R}_{25} \\ \hat{R}_{15} \end{bmatrix}, \tag{8-38}$$

As with the amplitude gain equations, the phase gain equations are also singular since the phase gains can only be determined to within a rotation factor. From a practical perspective, the use of the "arg(·)" poses problems due to the mapping discontinuity between $-\pi$ and π. To resolve these problems, we can arbitrarily define a reference phasor for the first super diagonal as the composite phasor resulting from summing the normalized phasors along the first super diagonal,

$$\rho_1 = e^{\iota\varphi_1} = \overline{e^{\iota\varphi_1}} = \frac{\sum_{k-1}^{K-1} \frac{\hat{R}_{k\,(k+1)}}{\left|\hat{R}_{k\,(k+1)}\right|}}{\left|\sum_{k-1}^{K-1} \frac{\hat{R}_{k\,(k+1)}}{\left|\hat{R}_{k\,(k+1)}\right|}\right|}. \tag{8-39}$$

If an amplitude gain adjustment has been applied to the sampled covariance matrix so that the magnitudes are approximately equal along any given super-diagonal, Eqn (8-39) may be approximated as,

$$\rho_1 \approx \frac{\sum_{k-1}^{K-1} \hat{R}_{k\,(k+1)}}{\left|\sum_{k-1}^{K-1} \hat{R}_{k\,(k+1)}\right|}. \tag{8-40}$$

In a similar fashion, we can define average phasors along each of the super diagonals such that

$$\rho_m \approx \frac{\sum\limits_{k-1}^{K-m} \hat{R}_{k\,(k+m)}}{\left|\sum\limits_{k-1}^{K-m} \hat{R}_{k\,(k+m)}\right|} \qquad m = 1, 2, \ldots, K-1. \tag{8-41}$$

Recognizing that—as with the amplitude gain estimation—that the $\hat{R}_{1K}$ element of the covariance matrix does not provide any information with respect to estimating the sensor phase gains, and using the specification for the reference phase along the first super diagonal, the a non-singular system of equations which avoids the problem of modulo arithmetic under the assumption that the sensor phase errors are small may be represented as

$$\boldsymbol{C}\begin{bmatrix}\varphi\\ \psi\end{bmatrix} = \boldsymbol{p} \tag{8-42}$$

where

$$\varphi = \begin{bmatrix}\varphi_0 \; \varphi_1 \; \cdots \; \varphi_{K-1}\end{bmatrix}^{\mathrm{T}}, \tag{8-43}$$

$$\psi = \begin{bmatrix}\tilde{\psi}_2 \; \tilde{\psi}_3 \; \cdots \; \tilde{\psi}_{K-2}\end{bmatrix}^{\mathrm{T}} \tag{8-44}$$

and

$$\boldsymbol{p} = \arg(\begin{bmatrix}\operatorname{diag}(\hat{\boldsymbol{R}}, 1)\rho_1^{\,*}\\ \operatorname{diag}(\hat{\boldsymbol{R}}, 2)\rho_2^{\,*}\\ \vdots\\ \operatorname{diag}(\hat{\boldsymbol{R}}, K-1)\rho_{K-1}^{\,*}\end{bmatrix}) \tag{8-45}$$

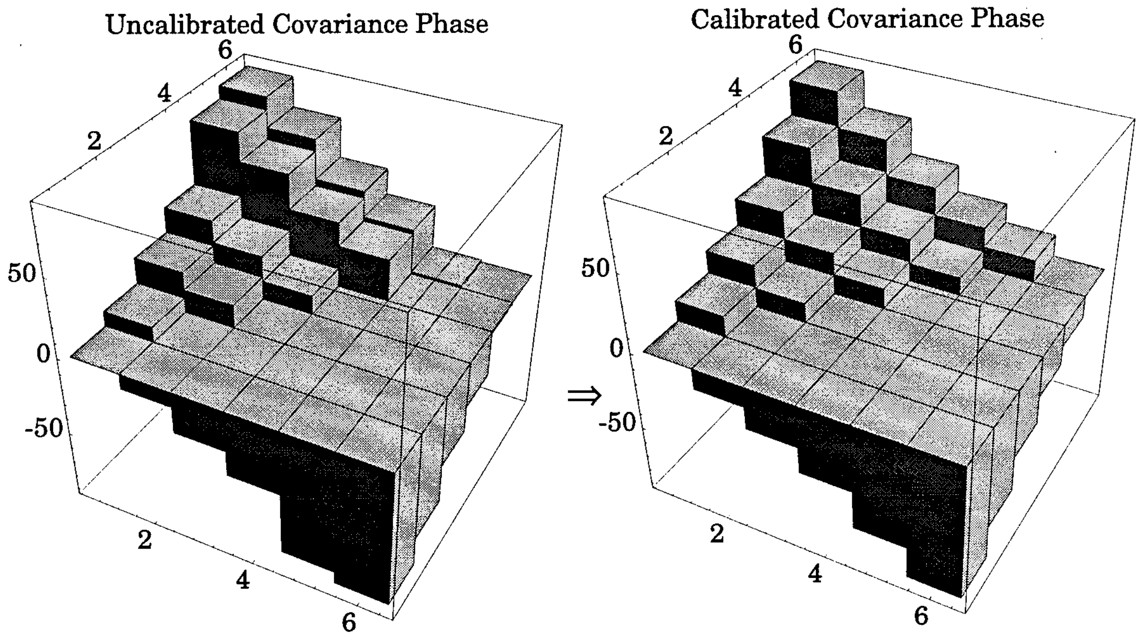
where the now estimated quantity $\tilde{\psi}_k$ indicates the phase shift of the k^{th} super-diagonal relative to the nominal average of the sampled covariance and its associated sensor gain errors. The matrix, $\boldsymbol{C}$, in Eqn (8-42) is the coefficient matrix resulting from deleting the column associated with the (now specified) first super-diagonal phase and the row associated with the $\hat{R}_{1K}$ element of the covariance. To illustrate, for the 5-element array case of Eqn (8-25), the structure of Eqn (8-42) becomes,

$$\begin{bmatrix} 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & -1 & 0 & 0 & 0 \\ 1 & 0 & -1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & -1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & -1 & 1 & 0 & 0 \\ 1 & 0 & 0 & -1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & -1 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} \psi_0 \\ \psi_1 \\ \psi_2 \\ \psi_3 \\ \psi_4 \\ \varphi_2 \\ \varphi_3 \end{bmatrix} = \arg\left(\begin{bmatrix} \mathrm{diag}(\hat{\boldsymbol{R}}, 1)\rho_1^{*} \\ \mathrm{diag}(\hat{\boldsymbol{R}}, 2)\rho_2^{*} \\ \mathrm{diag}(\hat{\boldsymbol{R}}, 3)\rho_3^{*} \end{bmatrix}\right), \qquad (8\text{-}46)$$

The least-squares solution to Eqn (8-42) is achieved via the pseudo-inverse of $\boldsymbol{C}$,

$$\begin{bmatrix} \varphi \\ \psi \end{bmatrix} = (\boldsymbol{C}^{\dagger}\boldsymbol{C})^{-1}\boldsymbol{C}^{\dagger}\boldsymbol{p} = \boldsymbol{C}^{\#}\boldsymbol{p}. \qquad (8\text{-}47)$$

The effect of phase calibration on an actual (as opposed to simulated) sampled covariance matrix is illustrated in Figure 8-10 for two strong broadband sources at DOAs of approximately ±7.5 degrees. As with the amplitude gain estimation, this approach is computationally efficient since the pseudo-inverse is a constant matrix dependent only upon the number of


Uncalibrated Covariance Phase

Calibrated Covariance Phase

Figure 8-10: Illustration of the Effect of Phase Gain Adjustment

elements in the sensor array; however, some complexity is involved in the arrangement of the measurement vector due to the need to avoid problems with modulo arithmetic.

8.4.4 Summary and Comments

Although conceptually simple and computationally efficient, the array calibration approach presented in this section *only* applies to the case wherein at least one *uncorrelated strong* source impinges on the sensor array. If the dominant sources are correlated, then the fundamental assumption of a Toeplitz structure will be violated; similarly, if only noise (or noise and a weak signal) is arriving at the array, the idealized Toeplitz structure will not be achieved.

In general, array calibration will not improve the detection performance since dyad multiplication corresponds to a subspace *rotation* as opposed to *warping* produced by a matrix multiplication (e.g., the colored noise whitening discussed in Section 8.3). The benefit of array calibration is to align the *actual* array topology (and associated manifold) with the assumed model—which improves the DOA estimation accuracy. This effect is demonstrated in Figure 8-11 which shows the Pisarenko root locations for a sequence of data set covariance matrices before and after calibration. In principle, true signal DOAs should be stable *and* reside on the unit circle with roots resulting from additive noise dispersed randomly over the complex-plane. In reality, the structure imposed by structured noise and propagation-related scattering results in stable roots which—although they do not reside on the unit circle—

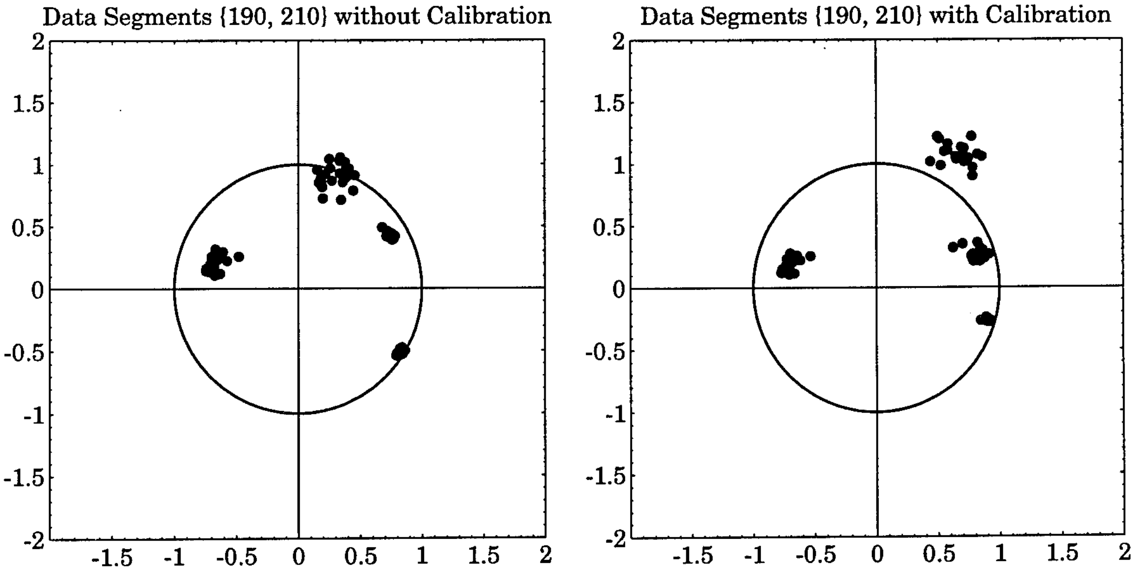


Figure 8-11: Array Calibration Effect on Pisarenko's Method Roots

preclude the use of information-theoretic eigenvalue analysis criteria for signal enumeration. Following calibration, the root-Tracker method is able to identify the roots associated with valid signals and accurately determine their DOAs. For the data set illustrated, the true roots should map into DOAs of ±7.5 degrees—which corresponds to the two roots nearest the unit circle.

8.4.5 Calibration and the Subspace Stability Techniques

Since the subspace stability methods (discussed in Chapter 7) exploit a sequence of sampled covariance matrices, it is natural to use this same sequence to estimate the sensor gains—thereby improving the gain estimation accuracy and confidence and providing some resilience against situations wherein the Toeplitz assumption is violated. Assuming he sensor gains are constant—albeit unknown—over the processing span leads to a measurement model of the form,

$$\tilde{\boldsymbol{g}}_n = \boldsymbol{g} + \boldsymbol{w}_n, \tag{8-48}$$

where $\tilde{\boldsymbol{g}}_n$ indicates the gain estimates derived from the n^{th} sampled covariance matrix and $\boldsymbol{w}_n$ is the perturbations in the estimate due to sensor noise and sampling effects. Assuming the perturbations are temporally and spatially white and inversely proportional to the received signal-to-noise ratio, and making the additional assumption that the additive noise power, σ^2, is constant and "small" permits the approximation,

$$\boldsymbol{w}_n = \frac{\sigma^2 \tilde{\boldsymbol{w}}}{\int_T \left(\sum_{j=1}^{J} |s_j(t)|^2 \right) dt} \approx \frac{\sigma^2 \tilde{\boldsymbol{w}}}{\int_T \left(\sum_{j=1}^{J} |s_j(t)|^2 \right) dt + \sigma^2} = \frac{\boldsymbol{w}}{A_n}, \tag{8-49}$$

where $A_n = a_{0_n}$ is the estimated signal power derived from the n^{th} covariance matrix and $\boldsymbol{w}$ is the complex-valued random variable normalized such that it has constant distribution across sampling epochs. We could adopt a weighted least squares estimation approach,

$$\hat{\boldsymbol{g}}(N) = \frac{\sum_{n=1}^{N} A_n \tilde{\boldsymbol{g}}_n}{\sum_{n=1}^{N} A_n} \tag{8-50}$$

which may also be expressed in the equivalent recursive form as,

$$\hat{\boldsymbol{g}}(N) = \frac{1}{1+\delta_N}\{\hat{\boldsymbol{g}}(N-1)+\delta_N\tilde{\boldsymbol{g}}_N\} \tag{8-51}$$

where,

$$\delta_N = \frac{A_N}{\sum_{N-1} A_n}. \tag{8-52}$$

Alternately, a Kalman filtering approach could be adopted; however, it should be noted that due to the approximation of the perturbation covariance matrix as,

$$\Lambda_n \approx A_n\boldsymbol{I} \tag{8-53}$$

there will be an artificial lower limit on the SNR (i.e., an upper limit on the assumed noise contributions) which will come into play in the event of low SNRs or the absence of a signal. In this unforced, constant gain system, the Kalman filtering equations are:

$$\hat{\boldsymbol{g}}(n) = \hat{\boldsymbol{g}}(n-1)+\boldsymbol{K}(n)\,[\tilde{\boldsymbol{g}}(n)-\hat{\boldsymbol{g}}(n-1)] \tag{8-54}$$

$$\boldsymbol{P}(n) = [\boldsymbol{I}-\boldsymbol{K}(n)]\,\boldsymbol{P}(n-1) \tag{8-55}$$

$$\boldsymbol{K}(n) = \boldsymbol{P}(n-1)\,[\Lambda_n+\boldsymbol{P}(n-1)]^{-1} \tag{8-56}$$

with the initial estimates of the gain and covariance being,

$$\hat{\boldsymbol{g}}_0 = \tilde{\boldsymbol{g}}_0 \tag{8-57}$$

and

$$\boldsymbol{P}(0) = \Lambda_0 = A_0\boldsymbol{I}. \tag{8-58}$$

Although the Kalman filtering equations have a simple form for this constant gain estimation scenario, they represent a considerable increase in processing relative to the recursive update equation of Eqn (8-51) and for some applications the benefits of Kalman filtering will not offset the increased computational burden or algorithmic complexity. In either case, care must be taken to avoid numerical round-off errors in the generation of the update gains, δ_N or $\boldsymbol{K}(n)$ during long duration processing.

9. Spatial Smoothing

The algorithms to this point presumed *incoherent* wavefronts arrive at the sensor array. If this assumption is violated so that *coherent* wavefronts are impinging upon the array, the subspace-based detection algorithms will underestimate the number of impinging *wavefronts*—instead returning the number of impinging *signals.*[1] Furthermore, for the coherent signals, the subspace DOA algorithms will either produce erroneous DOAs or fail to estimate the DOAs. Since coherent wavefronts may be generated via physical effects like multipath propagation from a common source or via similar waveforms from distinct sources, robust methods to determine the number of wavefronts and their associated directions is important to many applications.

For special array geometries—the most common being a uniform linear array—pre-processing the sampled array covariance matrix can obviate the problems of coherency and permit transparent application of the subspace algorithms. This technique, termed *spatial smoothing*, was first proposed in 1981 by Evans [29]. As with Schmidt's MUSIC, over the subsequent years variants were developed and the accuracy performance analyzed under a variety of scenarios.

In this chapter we explore the spatial smoothing from a significantly different viewpoint than used in the published literature in that we return to the vector subspace perspective implicit in Schmidt's seminal work. This powerful vantage point facilitates a new, intuitive, and aesthetic proof of the validity of spatial smoothing and its variants. Furthermore, it leads us to propose an new algorithm for the estimation of the directions of the impinging wavefronts.

In the following sections we review the problem coherent signals pose to the signal subspace algorithms. We then validate the spatial smoothing concept from the vector subspace perspective and show how the "classical" spatial smoothing techniques may be expressed in terms of the subspaces. Finally, we outline the proposed scheme to process a coherent scene which may have computational and performance advantages.

[1] Previously, we used the terms "wavefront" and "signal" interchangeably due to the assumption that all wavefronts represented different signal waveforms; however, for the purposes of this chapter, we must now draw the distinction that a given signal may arrive at the array via multiple wavefronts. In this case, the wavefronts are *coherent.* Thus, we draw the distinction that W wavefronts arrive at the array from J distinct waveforms/signals where, by definition, $J \leq W$.

9.1 Spatial Smoothing Assumptions

Much like the signal subspace algorithms discussed in Chapter 6, the spatial smoothing approach assumes an array topology such that sensor doublets (triplets, etc.) may be derived from the array. Furthermore, if forward-backward smoothing is to be used, the array must have a symmetric element spacing. Thus, for most practical purposes, we are restricted to a uniform, linear array structure. In this chapter we will assume a uniform linear array; however, the development and conclusions remain valid providing the array may be decomposed into similar, spatially distinct subarrays.

Consistent with general subspace processing, spatial smoothing also requires that the impinging wavefronts be narrowband. Note that even though several wavefronts may be arriving at an array via multipath from a single source, these wavefronts may not be coherent if the differential propagation delay is large enough. That said, *correlated* (partially coherent) wavefronts still pose a detection difficulty since the signal eigenvalues may not be much larger than those of the background noise. In this case, spatial smoothing helps to "spread" the vectors which define the signal subspace—thereby improving the detection of correlated signals.

9.2 The Problem with Coherent Wavefronts

To illustrate the effects of coherency, assume W distinct wavefronts from a single source signal, $s(t)$, arrive at a K-element array. In a slight variation of the development of Section 2.3, the signal received at the array elements may be expressed as,

$$\boldsymbol{x}(t) = \boldsymbol{D}\boldsymbol{a}s(t) \tag{9-1}$$

where $\boldsymbol{D}$ is the steering matrix which maps the wavefront impinging from the W DOAs onto the sensors of the array,

$$\boldsymbol{D} = \begin{bmatrix} 1 & \dots & 1 \\ \nu_1 & & \nu_W \\ \vdots & & \vdots \\ \nu_1^{K-1} & \dots & \nu_W^{K-1} \end{bmatrix} \tag{9-2}$$

with ν_w being the element-to-element phase shift of the w^{th} waveform,

$$\nu_w = e^{-\iota\varphi_w} = e^{-\iota 2\pi\delta\sin\theta_w}. \tag{9-3}$$

δ is the inter-element spacing (in units of wavelengths) and θ_w is the wavefront DOA. Since the waveforms are assumed to be coherent, the w^{th} waveform at the reference ("first") element is simply an attenuated and phase-shifted replica of the reference waveform, i.e., $s_w(t) = a_w s(t)$. Therefore, $\boldsymbol{a}$ may be viewed as the set of complex attenuations mapping the reference waveform onto the reference element,

$$\boldsymbol{a} = \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_W \end{bmatrix}. \tag{9-4}$$

Denoting the columns of $\boldsymbol{D}$ by $\boldsymbol{d}$—which corresponds to the mapping of the wavefront from the w^{th} direction onto the array elements and is commonly termed a *steering vector*—we see that a composite *mapping vector*, $\tilde{\boldsymbol{d}}$, may be defined as

$$\tilde{\boldsymbol{d}} = \boldsymbol{D}\boldsymbol{a} = a_1\boldsymbol{d}_1 + a_2\boldsymbol{d}_2 + \ldots + a_W\boldsymbol{d}_W. \tag{9-5}$$

Because of the Vandermonde structure of the steering vectors, the mapping vector *cannot* correspond to a valid steering vector—unless only one wavefront is present. In other words, a valid steering vector has a Vandermonde structure and a coherent mapping vector *cannot* have a Vandermonde structure; to illustrate, consider the two wavefront case,

$$\begin{bmatrix} a_1 + a_2 \\ a_1 v_1 + a_2 v_2 \\ a_1 v_1^2 + a_2 v_2^2 \\ \vdots \\ a_1 v_1^{K-1} + a_2 v_2^{K-1} \end{bmatrix} \neq (a_1 + a_2) \begin{bmatrix} 1 \\ \tilde{v} \\ \tilde{v}^2 \\ \vdots \\ \tilde{v}^{K-1} \end{bmatrix}; \qquad a_1, a_2 \neq 0. \tag{9-6}$$

This absence of a Vandermonde structure is exploited by the spatial smoothing algorithms to isolate the contributions of the coherent wavefronts while not affecting the contributions from incoherent wavefronts.

If we generalize the notation of Eqn (9-1) to permit a vector of J independent signals, $\boldsymbol{s}(t)$, and an array, $\boldsymbol{A}$, which maps these signals to the W distinct wavefronts, the signal received at the array elements, $\boldsymbol{x}(t)$, may be represented as,

$$\boldsymbol{x}(t) = \boldsymbol{D}\boldsymbol{A}\boldsymbol{s}(t) = \tilde{\boldsymbol{D}}\boldsymbol{s}(t). \tag{9-7}$$

In this general case, the j^{th} column (mapping vector) of the mapping matrix $\tilde{\boldsymbol{D}}$ maps the j^{th} signal onto the array elements. If a signal corresponds to a unique wavefront, the column has the structure of a steering vector; otherwise, this structure will be absent. The covariance matrix associated with the signal is therefore,

$$\boldsymbol{R} = \frac{1}{T}\int_T \boldsymbol{x}\boldsymbol{x}^\dagger\, dt = \frac{1}{T}\int_T \boldsymbol{D}\boldsymbol{A}\boldsymbol{s}(t)\boldsymbol{s}^\dagger(t)\boldsymbol{A}^\dagger\boldsymbol{D}^\dagger\, dt = \boldsymbol{D}\boldsymbol{A}\left(\frac{1}{T}\int_T \boldsymbol{s}(t)\boldsymbol{s}^\dagger(t)dt\right)\boldsymbol{A}^\dagger\boldsymbol{D}^\dagger = \boldsymbol{D}\boldsymbol{A}\boldsymbol{R}_s\boldsymbol{A}^\dagger\boldsymbol{D}^\dagger \quad (9\text{-}8)$$

where $\boldsymbol{R}_s$ is the signal covariance matrix. Note that the rank of $\boldsymbol{R}$ in Eqn (9-8) is determined by the rank of $\boldsymbol{R}_s$—i.e., the number of independent *signals* vice the number of impinging *wavefronts*. Assuming additive noise independent of the signals leads to a sampled covariance matrix of

$$\boldsymbol{R} = \boldsymbol{D}\boldsymbol{A}\boldsymbol{R}_s\boldsymbol{A}^\dagger\boldsymbol{D}^\dagger + \Sigma_n = \boldsymbol{D}\boldsymbol{A}\boldsymbol{R}_s\boldsymbol{A}^\dagger\boldsymbol{D}^\dagger + \sigma^2\boldsymbol{I} \quad (9\text{-}9)$$

where the latter relationship holds if the sensor noise is equi-powered, spatially independent, and zero-mean. Of course, this expression for the covariance matrix is idealized; however, it is sufficient for our current purposes. (Appendix A addresses the convergence behavior of the sampled covariance matrix as a function of the noise statistics and averaging interval.)

Figure 9-1 illustrates the effects of coherent wavefronts on the subspace-based DOA methods. Here two coherent wavefronts and one independent wavefront impinge on the array in a noise-

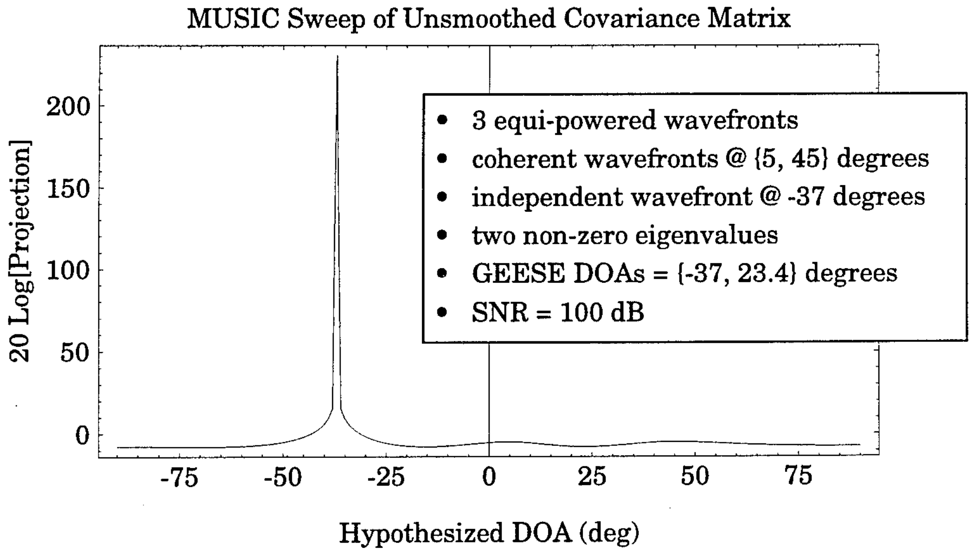


Figure 9-1: Spatial Response without Smoothing

free environment. Note that an eigenvalue analysis indicates that only two signals are present (rather than three wavefronts) and that the spatial spectrum indicates that only *one* valid signal is present—that of the independent wavefront; due to the coupling the coherent wavefronts, the coherent eigenvector does not correspond to a valid steering vector! However, given an appropriate array topology, it is possible to correctly isolate and identify the signal subspace.

9.3 The Validity of Spatial Smoothing

Evans [29] proposed partitioning the sensor array into N K_s-element similar subarrays and averaging the resulting covariance matrices to produce a "smoothed" covariance matrix. This is functionally equivalent to averaging principal subarrays along the main diagonal of the total array covariance matrix, i.e.,

$$\boldsymbol{R} = \begin{bmatrix} r_{11} & r_{12} & r_{13} & r_{14} & r_{15} & r_{16} \\ r_{21} & r_{22} & r_{23} & r_{24} & r_{25} & r_{26} \\ r_{31} & r_{32} & r_{33} & r_{34} & r_{35} & r_{36} \\ r_{41} & r_{42} & r_{43} & r_{44} & r_{45} & r_{46} \\ r_{51} & r_{52} & r_{53} & r_{54} & r_{55} & r_{56} \\ r_{61} & r_{62} & r_{63} & r_{64} & r_{65} & r_{66} \end{bmatrix} . \tag{9-10}$$

The validity of this subarray averaging has been proved in the literature via sequences of matrix algebra operations. In this section, we will adopt an apparently novel vector subspace perspective which, hopefully, provides additional insight into the spatial smoothing process.

For notational simplicity, the following sections will, in turn, address the structure of a single coherent and incoherent signal's mapping vector onto the subarray in the absence of additive noise.

9.3.1 Incoherent Wavefront and Spatial Smoothing

For an incoherent wavefront, the signal received at each of the N subarray's is related to signal at the reference element by,

$$\boldsymbol{x}_n(t) = \alpha_n \boldsymbol{d}s(t) = \tilde{\boldsymbol{d}}_n s(t) \tag{9-11}$$

where $\boldsymbol{d}$ is the K_s-element steering vector and α_n is a complex-valued phasor (unit magnitude) resulting from the propagation—or, equivalently, phase shift assuming negligible attenuation during the propagation across the array—from the reference element to the first element of the subarray. If the array is uniform linear, the steering vector will have the structure,

$$\boldsymbol{d} = \begin{bmatrix} 1 \\ \nu \\ \nu^2 \\ \vdots \\ \nu^{K_s - 1} \end{bmatrix}, \tag{9-12}$$

where ν is the element-to-element phase shift of the signal due to the wavefront propagation.

Since multiplication by a complex scalar does not change the orientation of a vector in C^K, from Eqn (9-11) we see that each of the subarray's mapping vector, $\tilde{\boldsymbol{d}}_n$, are aligned. Therefore, averaging of the subarray's covariance matrices results in the same covariance matrix as any single covariance matrix,

$$R^s = \frac{1}{N}\sum_{n=1}^{N} \boldsymbol{R}_n = \frac{1}{N}\sum_{n=1}^{N} \boldsymbol{x}_n \boldsymbol{x}_n{}^\dagger = \boldsymbol{d}\boldsymbol{d}^\dagger\, \overline{|s(t)|^2}\, \frac{1}{N}\sum_{n=1}^{N} |\alpha_n|^2 = \boldsymbol{R}_n. \tag{9-13}$$

In summary, other than reducing the number of array elements, spatial smoothing does not have an effect on the covariance matrix contributions of independent wavefronts.

9.3.2 Coherent Wavefronts and Spatial Smoothing

For W coherent wavefronts arriving at the array, the signal received at each of the N subarrays is related to the signal at the reference element, $s(t)$, by,

$$\boldsymbol{x}_n = \left(\sum_{w=1}^{W} \alpha_{w_n} a_w \boldsymbol{d}_w\right) s(t) = \left(\sum_{w=1}^{W} \tilde{a}_{w_n} \boldsymbol{d}_w\right) s(t) = \boldsymbol{D}\tilde{\boldsymbol{a}}_n s(t) = \tilde{\boldsymbol{d}}_n s(t) \tag{9-14}$$

where α_{w_n} is the phase shift due to array propagation to the first element of the subarray. As discussed earlier, the mapping vector, $\tilde{\boldsymbol{d}}_n$ cannot correspond to a valid steering vector. However, since the mapping vector is a linear combination of valid steering vectors, it *must* lie within the subspace defined by those steering vectors. Furthermore, since the propagation delay of each wavefront is a function of the wavefront DOA, each mapping vector must correspond to a *different* linear combination of the steering vectors. As a result, averaging N subarray covariance matrices,

$$R^s = \frac{1}{N}\sum_{n=1}^{N}\boldsymbol{R}_n = \frac{1}{N}\sum_{n=1}^{N}\boldsymbol{x}_n\boldsymbol{x}_n{}^\dagger = \boldsymbol{D}\left(\frac{1}{N}\sum_{n=1}^{N}\tilde{\boldsymbol{a}}_n\tilde{\boldsymbol{a}}_n{}^\dagger\right)\boldsymbol{D}^\dagger\,|s(t)|^2 = \boldsymbol{D}\tilde{\boldsymbol{A}}\tilde{\boldsymbol{A}}^\dagger\,\boldsymbol{D}^\dagger\,|s(t)|^2. \qquad (9\text{-}15)$$

By definition, $\boldsymbol{D}$ is a $K_s \times W$ matrix of full rank. $\tilde{\boldsymbol{A}}$ is a $W \times N$ matrix,

$$\tilde{\boldsymbol{A}} = \left[\boldsymbol{a}_1 \;\dots\; \boldsymbol{a}_N\right], \qquad (9\text{-}16)$$

and is of full rank since each $\tilde{\boldsymbol{a}}_n$ is unique due to the different wavefront propagations to the reference elements of the subarrays. As a result, the rank of the smoothed covariance matrix will be the minimum of the rank of $\tilde{\boldsymbol{A}}$ or $\boldsymbol{D}$. Hence, spatial smoothing will identify the wavefront subspace—providing at least W subarrays are averaged. If more subarrays are averaged, the additional covariance matrices are redundant and, therefore, do not change the rank of the smoothed covariance.

Examining Eqn (9-15) from a computational perspective, we see that we would like to "spread" the wavefront subspace as much as possible to improve the signal detection performance. Since the DOAs are defined by the scenario geometry, the only aspect of the formation of the smoothed covariance matrix under our control is the selection of the subarrays. A technique discussed later—forward-backward smoothing—accomplishes this objective.

Finally, we should note that the total energy contained within the smoothed covariance matrix (determined by either the trace or Frobenius norm) remains equal to that of any of the subarrays; however, this energy is now distributed over a subspace rather than a line in $\boldsymbol{C}^K$. Thus, we see the result of decreased large eigenvalues and increased small eigenvalues.

9.3.3 Additive Noise and Spatial Smoothing

By definition, the additive noise does not correspond to a DOA. Hence, the contribution of the noise to the smoothed covariance will simply be,

$$\boldsymbol{R}^s = \frac{1}{N}\sum_{n=1}^{N}\Sigma_n. \qquad (9\text{-}17)$$

If the noise is spatially uncorrelated, spatial smoothing serves to increase the effective number of noise samples averaged—thereby improving the convergence of the sampled noise covariance towards the asymptotic limit. Counteracting this, the effective reduction of array aperture leads to the steering vectors being less discernible from the noise. Typically, the net effect of classical spatial smoothing is to reduce the sensitivity of the subspace processing if all of the wavefronts are independent.

9.3.4 Forward-Backward Smoothing

To this point we have addressed averaging subarrays along the main diagonal of the sampled covariance matrix; this approach is commonly termed *forward smoothing*. To mitigate the reduction in array aperture associated with spatial smoothing, Pillai and Kwon [32] and Kwon [3] proposed a technique known as *forward-backward smoothing*. Basically, they realized that if the reference element of the sensor array is shifted from the "first" to the "last" element, the reverse-ordered vector of received signal values may be expressed as,

$$\boldsymbol{x}_b = \boldsymbol{D}_b \boldsymbol{a}_b s(t) \tag{9-18}$$

where for a uniform linear array the backwards steering matrix, $\boldsymbol{D}_b$ is

$$\boldsymbol{D}_b = \begin{bmatrix} 1 & \dots & 1 \\ \mathrm{v}_1^{-1} & & \mathrm{v}_W^{-1} \\ \vdots & & \vdots \\ \mathrm{v}_1^{-(K-1)} & \dots & \mathrm{v}_W^{-(K-1)} \end{bmatrix} = \boldsymbol{D}^* \tag{9-19}$$

and $\boldsymbol{a}_b$ is the complex-valued attentuation associated with each wavefront at the reference ("last") element. Noting that $\boldsymbol{D}_b = \boldsymbol{D}^*$, we see that upon complex-conjugation the received signal is,

$$\boldsymbol{x}_b^* = \boldsymbol{D}\tilde{\boldsymbol{a}}_b^* s(t) = \tilde{\boldsymbol{D}}_b s(t) \ . \tag{9-20}$$

Due to the assumption of a narrowband signal, we have incorporated the (approximately) 180 degree phase shift associated with conjugating the reference waveform, $s(t)$ into the attenuation and shifting coefficient, $\tilde{\boldsymbol{a}}_b^*$. Consistent with our earlier discussion on the effects of wavefront propagation, the $\boldsymbol{x}_b^*$ is a *different* linear combination of steering vectors than is the forward array's mapping vector—providing multiple coherent wavefronts are present. If an independent wavefront impinges on the array, the mapping vector of the "flipped" array, $\tilde{\boldsymbol{D}}_b$ will be aligned with the forward array's mapping vector, $\tilde{\boldsymbol{D}}$.

While it is possible to perform *backwards smoothing* on $\boldsymbol{x}_b^*$ analogous to the forward smoothing approach, the merger of the two viewpoints into *forward-backward smoothing* mitigates the array aperture reduction effects since for any given subarray size, the number of independent linear combinations of the steering vectors is doubled. Thus, for a given number of coherent wavefronts, the required subarray size is *generally* larger if forward-backward smoothing is employed.

Since the backwards sampled covariance matrix is equal to the conjugate of the forward covariance matrix which has been flipped over its anti-diagonal, i.e.,

$$\boldsymbol{R}^b = \boldsymbol{J}\boldsymbol{R}^{\mathrm{T}}\boldsymbol{J} = \boldsymbol{J}\boldsymbol{R}^*\boldsymbol{J} \tag{9-21}$$

where $\boldsymbol{J}$ is the permutation matrix,

$$\boldsymbol{J} = \begin{bmatrix} 0 & \dots & 0 & 0 & 1 \\ : & \dots & 0 & 1 & 0 \\ 0 & & & & : \\ 0 & 1 & 0 & \dots & : \\ 1 & 0 & 0 & \dots & 0 \end{bmatrix}. \tag{9-22}$$

The forward-backward smoothed covariance matrix may calculated by averaging appropriate sets of subarrays along the main diagonal of the forward covariance, $\boldsymbol{R}^f$, and the backwards covariance, $\boldsymbol{R}^b$. Thus,

$$\boldsymbol{R}^{fb} = \frac{1}{2N}\sum_{n=1}^{N}(\boldsymbol{R}_n^f + \boldsymbol{R}_n^b) = \frac{1}{2N}\sum_{n=1}^{N}\overline{\boldsymbol{R}}_n \tag{9-23}$$

where $\overline{\boldsymbol{R}}$ is the forward-backward smoothed array covariance matrix,

$$\overline{\boldsymbol{R}} = \boldsymbol{R}^f + \boldsymbol{R}^b. \tag{9-24}$$

As with the forward-smoothed case, forward-backward smoothing the sampled covariance will expand the rank of the covariance matrix—eventually producing a signal subspace which contains the includes the space of the steering vectors of the wavefronts.

9.3.5 Summary Comments

As discussed to date, spatial smoothing of the covariance matrix is a true pre-processing scheme since the resulting covariance may be used by the standard subspace detection and

DOA algorithms—providing $W_{\max}$ subarrays have been averaged where $W_{\max}$ is the maximum number of coherent wavefronts from a single source (as opposed to the *total* number of coherent wavefronts). Although any set of subarrays will satisfy the requirement to spread the coherent mapping vector(s), we intuitively expect that maximizing the difference between the linear combinations of steering vectors implicit in the array smoothing will improve the signal detection performance in the presence of noise. This is one of the reasons forward-backward smoothing tends to outperform either forward or backward smoothing.

To illustrate the effectiveness of spatial smoothing, consider Figure 9-2 which shows the spatial response after forward-backward spatial smoothing of the covariance matrix resulting from two coherent and one incoherent wavefronts. Unlike the results of processing the unsmoothed matrix demonstrated in Figure 9-1, the three wavefronts are both detectable and locatable.

Due to the generally better spreading as well as less aperture reduction, forward-backward smoothing should be used rather than either forward or backward smoothing. However, using the vector space concepts with which we proved the validity of spatial smoothing, the following section presents an apparently novel approach to the detection and DOA estimation which appears to offer improved accuracy and detection performance while being computationally efficient.

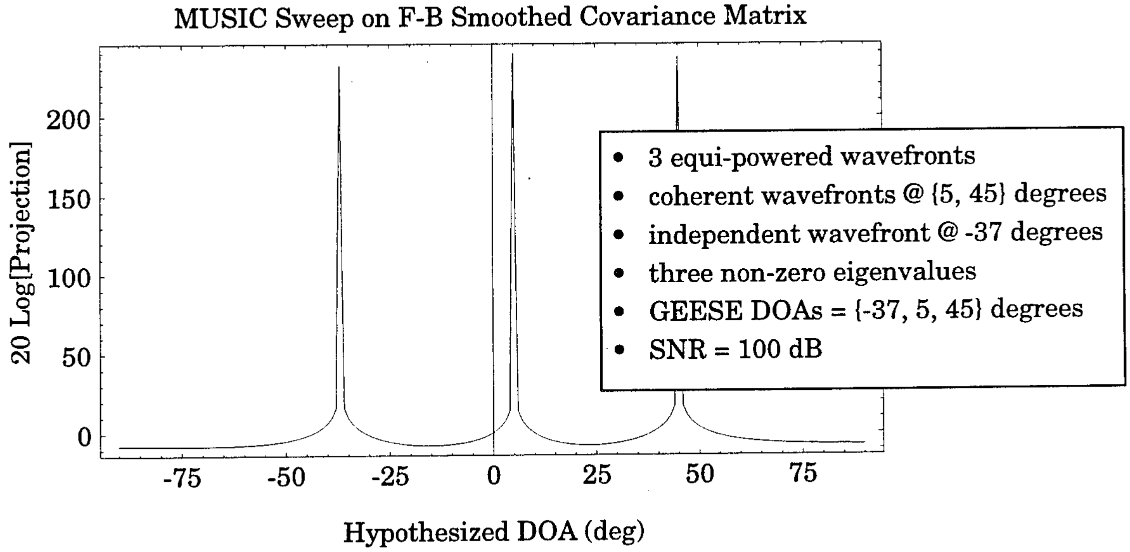


Figure 9-2: Spatial Response of a Coherent Scene after Smoothing

9.4 Proposed Improved Spatial Smoothing Algorithm

Although the classical spatial smoothing approach is effective, there are several undesirable aspects related to using submatrices of the sampled covariance matrix:

- Coherent signals are more easily *detected* when using the *unsmoothed* covariance matrix due to the consolidated energy of the wavefronts. As discussed in Chapter 4, determination of the number of signals is difficult since the typical assumptions are generally violated.
- There is no obvious way to determine when "enough" spatial smoothing has been applied other than examining the changes in the smoothed covariance eigenvalues following successive layers of smoothing. Thus, signal detection again becomes an issue.
- While spatial smoothing does not cause the noise level to increase, the aperture reduction effectively decreases the signal information—thus, decreasing the effective SNR and decreasing the accuracy of the determination of the number of signals as well as the DOA estimates.

In this section we briefly outline two recently proposed approaches to resolving coherent signals: the *signal eigenvector method* (SEM) and *postsmoothing* as well as a new approach which builds on the vector subspace concepts. This *steering vector subspace* (SVS) approach appears to offer computational advantages relative to the prior methods

9.4.1 Notation

If we denote the eigendecomposition of the sampled covariance matrix, $\boldsymbol{R}$, as

$$\boldsymbol{R} = \boldsymbol{B}\Lambda\boldsymbol{B}^{\dagger}, \tag{9-25}$$

where the columns of the $K \times K$ matrix $\boldsymbol{B}$ are the eigenvectors associated with the diagonal matrix of *ordered* (largest-to-smallest) eigenvalues, Λ. Analyzing the (real-valued) eigenvalues permits estimating the number of impinging *signals* (vice wavefronts), J. Using this estimate, the eigenvectors may be partitioned into those associated with the signals, $\boldsymbol{B}_s$, and noise (i.e., orthogonal subspace), $\boldsymbol{B}_o$,

$$\boldsymbol{B} = \left[\boldsymbol{B}_s\ \boldsymbol{B}_o\right]. \tag{9-26}$$

Similarly, we can partition the eigenvalues into a $J \times J$ matrix of signal eigenvalues, Λ_s, and a $(K-J) \times (K-J)$ matrix of noise eigenvalues, Λ_o, so that,

$$\Lambda = \begin{bmatrix} \Lambda_s & 0 \\ 0 & \Lambda_o \end{bmatrix}. \tag{9-27}$$

As noted in Chapter 4, the signal eigenvectors will be "large" relative to the noise eigenvalues. The partitioning of the sampled covariance matrix into a signal subspace and a noise subspace is fraught with peril; however, for the sequence of this chapter we will presume that the number of signals has been accurately estimated.

9.4.2 Signal Eigenvector Method (SEM)

Cadzow, *et al* [31] recognized that the signal eigenvectors are a linear combination of the steering vectors. Essentially, he proposed that a matrix be formed from the signal eigenvectors,

$$\Gamma = \boldsymbol{B}_s \boldsymbol{B}_s^{\dagger} \tag{9-28}$$

and that this approximation to the covariance matrix be smoothed and the result, $\boldsymbol{S}^{fb}$, be used in the subspace processing

9.4.3 Postsmoothing (Krim & Proakis)

Krim and Proakis [35] proposed to develop a reduced rank approximation to the sampled covariance matrix, $\hat{\boldsymbol{R}}$,

$$\hat{\boldsymbol{R}} = \boldsymbol{B}_s \Lambda_s \boldsymbol{B}_s^{\dagger}, \tag{9-29}$$

which would be spatially smoothed and the result processed using standard subspace detection and DOA estimation methods. This techniques was named *postsmoothing* because the smoothing is applied after an initial eigendecomposition as opposed to the "classical" approach (which they termed *presmoothing*) wherein the smoothing is done directly on the sampled covariance matrix.

Krim and Proakis contended that postsmoothing outperformed both SEM and presmoothing for most situations and that presmoothing generally outperformed SEM. This result is intuitively agreeable from a vector subspace perspective; working with the reduced rank approximation should provide improved performance from a vector subspace perspective due to the elimination of the identified noise subspace. The subsequent spatial smoothing will spread the noise energy attached to the over an expanded subspace—thereby improving the DOA estimation accuracy of the subsequent processing. The relative performance of the SEM approach may be an artifact of the specific simulation cases chosen by Krim and Proakis since using the eigenvectors rather than a reduced rank approximation should improve the DOA estimation performance in a mixed signal strength environment.

9.4.4 Spatial Smoothing using the Eigenvectors

Although Krim and Proakis formed the reduced rank approximation to the sampled covariance matrix and *then* performed the spatial smoothing, it should be noted that the smoothed covariance matrix may be achieved directly from the eigenvectors. If the array is partitioned into N similar subarrays then the corresponding eigenvectors of that subarray are simply the corresponding elements of the corresponding elements of the sampled covariance matrix' eigenvectors. Thus, if we adopt the notation,

$$\boldsymbol{B} = \begin{bmatrix} b_{11} & \cdots & b_{1K} \\ \vdots & & \vdots \\ b_{K1} & \cdots & b_{KK} \end{bmatrix} \tag{9-30}$$

where K is the size of the overall array and the columns of $\boldsymbol{B}$ correspond to the eigenvectors of the array, for K_s-element subarrays the n^{th} subarray's signal eigenvectors are

$$\boldsymbol{B}_{s_n} = \begin{bmatrix} b_{n1} & \cdots & b_{nJ} \\ \vdots & & \vdots \\ b_{(n+K_s)1} & \cdots & b_{(n+K_s)J} \end{bmatrix}. \tag{9-31}$$

In this case, reduced rank forward smoothing could be achieved by,

$$\hat{\boldsymbol{R}}^f = \frac{1}{N}\sum_{n=1}^{N} \boldsymbol{B}_{s_n}\Lambda_s\boldsymbol{B}_{s_n}{}^{\dagger} \tag{9-32}$$

or the reduced rank forward-backward smoothing via,

$$\hat{\boldsymbol{R}}^{fb} = \frac{1}{2N}\sum_{n=1}^{N} (\boldsymbol{B}_{s_n}\Lambda_s\boldsymbol{B}_{s_n}{}^{\dagger} + \boldsymbol{B}_{s_n}{}^{*}\Lambda_s\boldsymbol{B}_{s_n}^{\mathrm{T}}). \tag{9-33}$$

Although smoothing using the reduced rank approximation of Eqn (9-29) would be preferred for computational reasons over the approach represented by either Eqn (9-32) or Eqn (9-33), this formulation provides a little more insight into the traditional subarray averaging approach since we see how the signal eigenvectors—which must be linear combinations of the steering vectors—are used to define a subspace which spans the wavefront subspace. From our earlier discussions, we realize that if coherent wavefronts are present, the effect will be to expand the dimension of the signal subspace to the number of wavefronts.

9.4.5 Steering Vector Subspace Method (SVS)

Although neither Cadzow or Krim and Proakis addressed the improved detection performance resulting from using the unsmoothed covariance matrix, using the reduced rank approximation effectively lowers the noise floor which makes determining the number of wavefronts from the smoothed covariance matrix easier. On the negative side, both approaches implicitly require two eigendecompositions—an initial one to determine the signal subspace and another following the spatial smoothing to determine the desired *wavefront* subspace. Thus, there is a potentially significant computational burden imposed by these methods.

From our vector subspace perspective, we see that spatial smoothing may be employed to expand the spatial subspace to that of the wavefront subspace—therefore, the intermediate step of forming a smoothed covariance matrix is unnecessary. Let us define a $K_s \times JN$ matrix, $\boldsymbol{U}$, formed from selected elements of the signal eigenvectors,

$$\boldsymbol{U} = \left[\tilde{\boldsymbol{B}}_1 \ldots \tilde{\boldsymbol{B}}_N\right], \tag{9-34}$$

where $\tilde{\boldsymbol{B}}_n$ is comprised of the rows of the signal eigenvectors, $\boldsymbol{B}_s$, corresponding to the K_s-element subarray, J is the number of detected signals, and N is the number of subarrays. Of course, for optimal spread of the wavefront subspace and maximum resolving power, we should implement forward-backward smoothing—i.e., some of the subarrays should be the "flipped" conjugate of a forward subarray.

In the absence of noise, $\boldsymbol{U}$ will be a rank $\min(K_s, W, JN)$ matrix where W is the *total* number of wavefronts impinging upon the array. With additive noise perturbing the subspace, with a probability of one (w. p. 1) $\boldsymbol{U}$ will be of full rank. Just as with the original signal detection, the energy contained within the "noise subspace" should be "small" and distinguishable from the contributions of the wavefronts. Using a Gram-Schmidt orthogonalization, we can efficiently construct an orthogonal basis. In the absence of noise, we could use this basis directly in the DOA estimation; however, in general, we seek an *ordered* (by power) orthogonal basis set to facilitate the distinction between the signal and noise subspaces—which is note normally provided by Gram-Schmidt. However, due to the computational efficiency of the Gram-Schmidt procedure, we need to explore formulations which extract the desired basis set.

The singular value decomposition of $\boldsymbol{U}$ can provide this ordered basis set at the cost of some overhead since we would not use the "short space" of the decomposition.

We should also note that the definition of $\boldsymbol{U}$ in Eqn (9-34) is similar to that used by Cadzow in the SEM. An alternate formulation weighting the subarray eigenvectors by their associated signal strength may be preferable if the relative performance of the reduced rank approximation and SEM of Krim and Proakis holds.

Finally, we should recognize that our goal in the DOA estimation problem is to find the *non-orthogonal* basis set defined by the steering vectors. Computational efficiency and repeatability has mandated techniques based upon orthogonal decompositions; however, there may be conceptual as well as detectability benefits in adopting the non-orthogonal perspective.

In summary, the vector subspace perspective has identified some algorithmic and analysis issues which should be explored:

- Identify an efficient algorithm to extract the wavefront subspace from $\boldsymbol{U}$ or $\boldsymbol{V}$ based upon a Gram-Schmidt orthogonalization;
- Define an algorithm which exploits the SVD to identify the wavefront subspace and compare its performance to those derived from the Gram-Schmidt orthogonalizations;
- Determine the relative merits of using the subarray eigenvectors ($\boldsymbol{U}$) vice weighted eigenvectors ($\boldsymbol{V}$) in the DOA estimation. By extension, this will assess the performance claims of Krim and Proakis.
- Consider the implications of a non-orthogonal representation of the wavefront subspace and the associated signal detection and DOA estimation accuracy performance.

10. Concluding Remarks

10.1 Overview and Summary

Herein we have reviewed a variety of array covariance matrix-based approaches for detecting and estimating the directions-of-arrival (DOAs) of impinging wavefronts; those addressed include:

Signal Enumeration	• sequential testing • information-theoretic criteria – AIC – MDL – MIC
DOA Estimation	• direct-mapping techniques – split-aperture – full-aperture • inverse-mapping techniques – maximum entropy – minimum variance • orthogonal subspace methods – MUSIC – root-MUSIC – Pisarenko's method – minimum-Norm • signal subspace (matrix-shifting) methods – ESPRIT – TLS-ESPRIT – GEESE

The subspace methods have the attractive ability to simultaneously locate *multiple* impinging signals—*providing* an accurate signal enumeration and associated partitioning of the sampled covariance matrix into signal and noise subspaces is possible. Unfortunately, applications such as sonar array processing frequently violate the subspace processing assumptions of:

- narrowband signals,

- identical and accurately calibrated sensors,
- known noise covariance structure,
- stationary source-array geometries, and
- no wavefront dispersion during wavefront propagation—i.e., point sources.

The primary implication of violating these assumptions is that signal enumeration via information-theoretic criteria is invalidated. These criteria examine the eigenvalues resulting from an eigendecomposition of the sampled covariance matrix under the assumption of *i.i.d.* additive Gaussian sensor noise and calibrated array elements. An accurate signal enumeration is essential to the subspace partitioning required by the subspace DOA estimation algorithms. Since this effort was motivated by a desire to apply subspace processing techniques to real-time in-water sonar signal processing, a paradigm was identified which relaxed the restrictions imposed by the signal enumeration processing—the *subspace stability methods* presented in Chapter 7.,These approaches exploit the *a priori* knowledge of the array manifold and an assumption of continuously transmitting sources.

Under the assumption of continuously transmitting sources, the signal subspace will be temporally stable. Two computationally viable algorithms which are applicable for uniform linear arrays were identified and their performance against in-water data demonstrated. Both the *root-Tracker* and *Subspace Stability Exploitation Tracker* (SSET) techniques use multiple-target tracking (MTT) techniques coupled with traditional subspace-based DOA estimation approaches to enumerate and characterize signals. As was noted, the subspace stability paradigm may be exploited by other techniques such as pattern recognition; the target tracking approach was adopted because of its suitability to on-line, real-time, sequential data processing.

An adaptive array calibration algorithm was also developed (and presented in Section 8.4) which is suitable for on-line processing. This computationally efficient approach complements the root-Tracker and SSET approaches and increases their accuracy under the assumption of uncorrelated wavefronts

Correlated wavefronts pose a problem to subspace algorithms since the size of the signal subspace will correspond to the number of impinging signals rather than wavefronts. Fortunately, a pre-processing technique known as "spatial smoothing" can expand the signal subspace to the desired dimension. In Section 9.3, a novel proof of the validity of spatial smoothing is presented using a vector space perspective. This viewpoint suggests new approaches for coherent signal processing.

10.2 General Comments and Open Issues

Accurate signal enumeration is the most difficult aspect associated with applying subspace methods to in-water sonar array processing due to the low data rates and frequently violated processing assumptions. Sonar processing is often further complicated by dynamic source-array geometries. As a result, the root-Tracker and SSET represent a significant advance since they permit source identification and tracking while demanding relatively few computational resources. While these approaches can incorporate a variety of the subspace-based DOA estimation algorithms (e.g., ESPRIT, GEESE, root-MUSIC, etc.), the matrix-shifting algorithms appear to be most suited for practical implementation due to their computational efficiency and resistance to array calibration errors.

Since this research was motivated by the in-applicability of the classical assumptions to some realistic signal processing environments, perturbation analyses and extensive simulation analyses have not been included. Such analyses would be useful in conjunction with development of appropriate perturbation models to facilitate the optimization of the signal processing and data association aspects of the subspace stability methods. An obvious area for such an effort would be the GEESE algorithm under varying enumeration hypotheses.

Additional recommendations for further research on the subspace stability methods and coherent signal processing are presented in Section 7.7 and Section 9.4, respectively.

Appendix A. Covariance Matrix Statistics

Due to finite sampling intervals, the reality is that we will not obtain the idealized matrix of Eqn (2-24); however, we can hopefully integrate over enough samples to make a reasonable convergence on this form and we shall hope that we can eat the introduced error. Thus, in general,

$$\hat{\boldsymbol{R}} \neq \boldsymbol{R} = E\{\boldsymbol{x}\boldsymbol{x}^{\dagger}\} = E\{\boldsymbol{D}\boldsymbol{s}\boldsymbol{s}^{\dagger}\boldsymbol{D}^{\dagger} + \boldsymbol{D}\boldsymbol{s}\boldsymbol{n}^{\dagger} + \boldsymbol{n}\boldsymbol{s}^{\dagger}\boldsymbol{D}^{\dagger} + \boldsymbol{n}\boldsymbol{n}^{\dagger}\}, \tag{A-1}$$

where $\hat{\boldsymbol{R}}$ is the *estimated* covariance matrix derived from the sampled signal stream. Previously, we made the argument that zero-mean noise which was independent of the signal resulted in $E\{\boldsymbol{D}\boldsymbol{s}\boldsymbol{n}^{\dagger} + \boldsymbol{n}\boldsymbol{s}^{\dagger}\boldsymbol{D}^{\dagger}\} \to 0$, i.e., the cross-product terms *eventually* drop out and, furthermore, if we assume that the noise is spatially independent—a mathematically atheistic assumption albeit suspect in the presence of flow noise—then the noise covariance *eventually* converges to a diagonal matrix. In this section, we examine the statistics of that convergence as well as the covariance matrix.

A.1 Assumptions

The key assumptions in the derivations of this chapter are:

- Narrowband signals impinge upon the array;
- The signals are band-limited white Gaussian with a bandwidth of B_s;
- The sensor noise is spatially uncorrelated;
- The noise is band-limited white Gaussian with a bandwidth B; and,
- The sampled covariance matrix is derived by averaging over a time span of T which comprises N samples.
- Signals and noise are analytic with power spectral densities of σ_s^2 and σ_n^2, respectively.

If the signals cannot be treated as narrowband, then the modelling of the signal contribution to the covariance matrix as $\boldsymbol{D}\boldsymbol{s}\boldsymbol{s}^{\dagger}\boldsymbol{D}^{\dagger}$ in Eqn (A-1) is invalid as are the algorithms and analyses presented in this document. If the signal is deterministic, then the structure of $\boldsymbol{D}\boldsymbol{s}\boldsymbol{s}^{\dagger}\boldsymbol{D}^{\dagger}$ is also deterministic; however, the statistics of the cross-product terms will remain the same as in the random signal case.

The spatially uncorrelated noise is assumed to be band-limited white Gaussian. Although the derivation here permits differing ambient sensor noise variances, most of the detection and DOA estimation algorithms discussed in this document presume similar statistics at each sensor.

Finally, it is presumed that the sampled covariance matrix is the result of averaging the instantaneous covariances over a time span of T. Thus,

$$\boldsymbol{R} = \frac{1}{T}\int_{T}\boldsymbol{R}_t(t)dt \qquad \text{(A-2)}$$

or, if we are operating in the discrete domain with N samples, the normalized sampled covariance matrix is given by

$$\boldsymbol{R} = \frac{1}{N}\sum_{n=1}^{N}\boldsymbol{R}_n \qquad \text{(A-3)}$$

where $\boldsymbol{R}_n$ is the epoch covariance matrix.

A.2 General Probability Background and Comments

This section presents some of the background material common to the following sections. These include the probability density functions of the band-limited Gaussian noise, the effective correlation time and number of independent samples during the averaging span, and the simplifying contributions of the central limit theorem for modelling the statistics of the ensemble average.

A.2.1 Probability Density Functions

The random waveforms are assumed to be complex-valued Gaussian; hence, at any sampling epoch the *probability density function* (pdf) of the sample is the product of the real and imaginary components of the density function which are each assumed to have a real-valued Gaussian distribution,

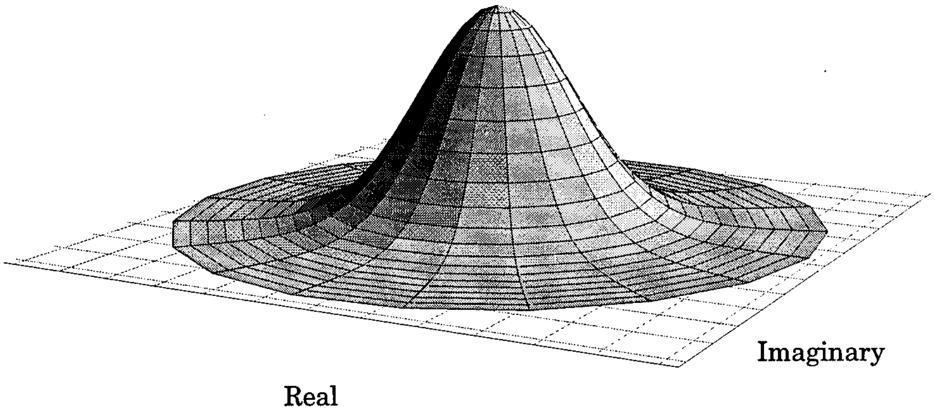


Figure A-1: Plot of the Complex-valued Gaussian Density Function

$$f_{\mathrm{Re}(z)} = f_x = \frac{1}{\sigma\sqrt{\pi}} e^{-(x-\mu)^2/\sigma^2} \qquad \text{(A-4)}$$

and,

$$f_{\mathrm{Im}(z)} = f_y = \frac{1}{\sigma\sqrt{\pi}} e^{-(y-\mu)^2/\sigma^2}. \qquad \text{(A-5)}$$

Defining the complex-valued variable z as $z = x + \iota y$ and assuming that x and y are independent and zero-mean ($\mu = 0$), leads to a joint density distribution of

$$f_z = f_x f_y = \frac{1}{\pi\sigma^2} e^{-(x^2+y^2)/\sigma^2}. \qquad \text{(A-6)}$$

This distribution is illustrated in Figure A-1. In the following sections, we will use this pdf definition to derive the mean and standard deviations of the components of the covariance matrix described by Eqn (A-1). Note that the pdf is rotationally invariant so that it could equivalently be described in (r, θ) coordinates with the radius, r, having a Rayleigh distribution and the angle, θ, uniformly distributed from 0 to 2π. Thus, the pdf for r is

$$f_r = \frac{2r}{\sigma^2} e^{-r^2/\sigma^2}. \qquad \text{(A-7)}$$

Note that the definitions of the density functions in Eqn (A-3)–Eqn (A-4) use a definition of σ derived from the noise power. The standard deviation of the Gaussian distributions for x and y are related to the noise power by,

$$\sigma_x = \sigma_y = \frac{\sigma}{\sqrt{2}}. \tag{A-8}$$

Because the Rayleigh distribution is defined based upon the characteristics of the underlying Gaussians, in the subsequent developments we will use,

$$\sigma_r = \frac{\sigma}{\sqrt{2}}. \tag{A-9}$$

A.2.2 Correlation Time and the Number of Independent Samples

Assuming an ideal band-limited Gaussian signal is equivalent to assuming that white noise is processed by a rectangular filter of bandwidth β prior to entering the system. In this case, it can be shown [65], that the autocorrelation function is simply a sinc function, i.e.,

$$R(\tau) = \sigma^2 \operatorname{sinc}(\tau\beta), \tag{A-10}$$

where σ^2 is the noise power within the bandwidth β. Under these assumptions, samples separated by exactly $1/\beta$ are uncorrelated and, therefore, independent. For a general signal distribution, the process correlation time is defined as the inverse of the effective bandwidth, i.e.,

$$T_{\text{cor}} = \frac{1}{\beta_e} = \frac{\int_{-\infty}^{\infty} |R(\tau)|^2 d\tau}{R^2(0)} \tag{A-11}$$

where here the effective bandwidth is equal to the actual bandwidth. Samples closer than T_{cor} will probably be correlated while those further away will be uncorrelated. Using this criterion, and assuming the data is averaged over a time span of T leads us to conclude that during that time span there will be approximately BT independent samples. Thus, we can define an effective (fractional) data set size of

$$N_{\text{eff}} = BT + 1 \tag{A-12}$$

where the "1" allows for the initial sample. If the raw data set of N samples is derived by sampling at a data rate of f_S, the effective data set size is related to the raw data size via,

$$N_{\text{eff}} = \frac{B}{f_S}(N-1) + 1. \tag{A-13}$$

A.2.3 The Central Limit Theorem

Although the density function of the elements of the *averaged* covariance matrix will not be Gaussian due to the processing, by virtue of the central limit theorem, the ensemble distribution will approach that of a Gaussian. Thus, for reasonably large averaging intervals—or, more precisely, a large time-bandwidth product—assuming a normal distribution is appropriate. Of course, the caveat applies that the "tails" of the distribution will approach a Gaussian model slower than will the region near the mean.

A.3 The Auto-Correlation Statistics

The main diagonal elements of the $\boldsymbol{s}\boldsymbol{s}^{\dagger}$ and $\boldsymbol{n}\boldsymbol{n}^{\dagger}$ components of Eqn (A-1) are the averaged auto-correlation of the random variables. In this section we will examine the *instantaneous* statistics associated with a given sampling epoch as well as the statistics of the ensemble average which contributes to the sampled covariance matrix used by the detection and DOA-estimation algorithms. Note that—by construction—these diagonal terms are real-valued.

A.3.1 The Autocorrelation Statistics at a Given Epoch

Defining z as a generic random variable (r.v.) representing either the j^{th} signal or the noise appearing at the k^{th} array element and assuming it has the pdf of Eqn (A-4), the expected value for any single *epoch* (sample) would be

$$E(zz^{\dagger}) = E(r^2) = \int_{-\infty}^{\infty} r^2 f_r dr = \sigma^2. \qquad \text{(A-14)}$$

Equivalently, we could use,

$$E(zz^{\dagger}) = E(x^2+y^2) = \int_{-\infty}^{\infty} (x^2+y^2) f_x f_y dxdy = \sigma_x^2 + \sigma_y^2 = \sigma^2, \qquad \text{(A-15)}$$

which is consistent with our knowledge that the power spectral density of a zero-mean Gaussian random variable is equal to its variance. Similarly, the second moment of the sample value at a given epoch is,

$$E(zz^{\dagger}\,(zz^{\dagger})^{\dagger}) = E(r^4) = \int_{-\infty}^{\infty} r^4 f_r dr = 2\sigma^4, \qquad \text{(A-16)}$$

or, in Cartesian coordinates,

$$E(zz^{\dagger}\,(zz^{\dagger})^{\dagger}) = E((x^2+y^2)\,(x^2+y^2)) = E(x^4+2x^2y^2+y^4) = E(x^4)+E(x^2y^2)+E(y^4) \quad \text{(A-17)}$$

which simplifies to the same result,

$$E(zz^{\dagger}\,(zz^{\dagger})^{\dagger}) = 3\sigma_x^4+2\sigma_x^2\sigma_y^2+3\sigma_y^4 = 2\sigma^2 \quad \text{(A-18)}$$

so that the standard deviation of the sample power at a given epoch is

$$\sigma_{|z|^2} = \sigma^2. \quad \text{(A-19)}$$

A.3.2 The Statistics of the Ensemble Average of the Autocorrelation

The *average* of the sum of the squares of M i.i.d. Gaussian random variables has the form[1] of a *chi-squared* density function with M *degrees of freedom*, χ_M^2. If we define c to be the sum of the squares of x which is a Gaussian r.v. with a standard deviation of σ_x, i.e.,

$$c = \sum_{m=1}^{M} x_m^2. \quad \text{(A-20)}$$

Alternately, we could define our random variable as the *average* (normalized) sum of the squares, i.e.,

$$\alpha = \frac{1}{M}\sum_{m=1}^{M} x_m^2. \quad \text{(A-21)}$$

The density function of the average, α, has the form [65],

$$f_{\chi_M^2}(\alpha) = \frac{(\alpha/(\sigma_x^2/M))^{M/2-1}}{2^{M/2}(\sigma_x^2/M)\,\Gamma(M/2)}e^{-\alpha/2(\sigma_x^2/M)} \quad \text{(A-22)}$$

where $\Gamma(\cdot)$ is the gamma function,[2]

[1] Note that the density function and the associated moments are not *quite* the Chi-squared distribution. The χ_N^2 distribution is usually defined as the sum of the squares of a Gaussian with a unitary standard deviation.

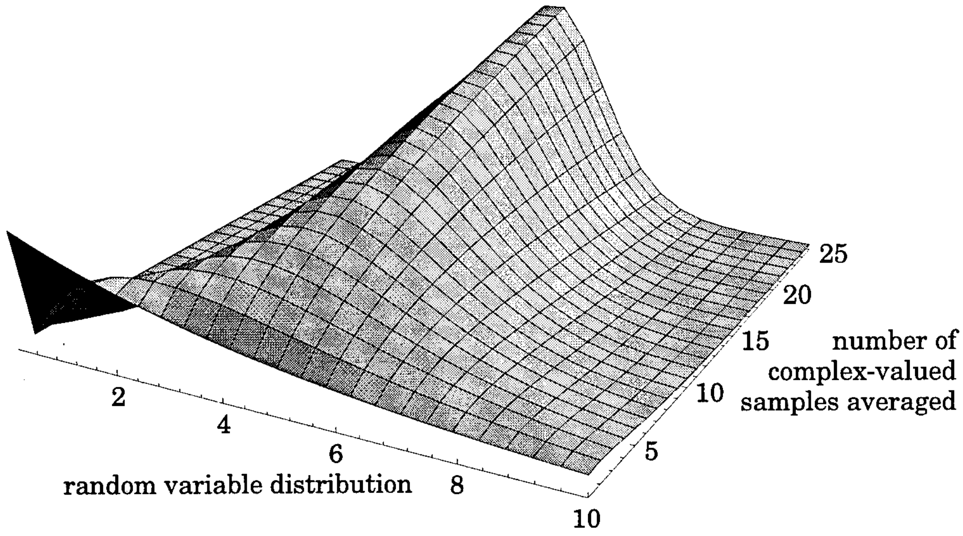


Figure A-2: Behavior of the Normalized χ^2_N Density vs. Sample Size ($\sigma^2_x = 5$)

$$\Gamma(\gamma) = \int_0^{\infty} u^{\gamma-1} e^{-u} du \qquad \text{(A-23)}$$

The behavior of this density function is plotted in Figure A-2. Note the transition from a Rayleigh distribution for $M = 2$ (corresponding to a *single complex-valued* sample) towards the Gaussian distribution predicted by the central limit theorem as the number of samples increases.

The mean and variance of the χ^2_N distribution is

$$E(\chi^2_M) = M\sigma^2_x \qquad \text{(A-24)}$$

and,

[2] As an aside, recall that for half-integer values of γ, the gamma function reduces to [67],

$$\Gamma(\frac{2m+1}{2}) = \frac{(2m)!}{m!}2^{-2m}\sqrt{\pi} \qquad m = 0, 1, 2, \ldots \qquad \text{(0-1)}$$

while for integer values it is simply related to the factorial,

$$\Gamma(m+1) = m! \qquad m = 0, 1, 2, \ldots \qquad \text{(0-2)}$$

$$\mathrm{Var}(\chi_M^2) = 2M\sigma^4 \tag{A-25}$$

while the mean and variance of the average is simply,

$$E(\alpha) = \sigma_x^2 \tag{A-26}$$

and,

$$\mathrm{Var}(\alpha) = \frac{2}{M}\sigma_x^4. \tag{A-27}$$

As we can see upon appropriate substitution, the distribution for the single epoch sample of the covariance matrix in Eqn (A-14) follows a two degree of freedom chi-squared distribution due to the independence of the real and imaginary components of the sample. As a result, the number of samples is twice the effective data size N_{eff}, derived in Section A.2.2. If we view the sampled N data points as being composed of N/N_{eff} duplicate sets of N_{eff} *complex-valued* samples, then the averaged autocorrelation expected value is

$$E(\alpha) = \frac{1}{N_{\mathrm{eff}}}E(\chi_N^2) = \frac{1}{N_{\mathrm{eff}}}(2N_{\mathrm{eff}})\sigma_x^2 = \sigma^2 \tag{A-28}$$

where we recall that $\sigma_x = \sigma/\sqrt{2}$. Similarly, the variance of the ensemble average is

$$\mathrm{Var}(\alpha^2) = \mathrm{Var}(\chi_N^2) = \left(\frac{1}{N_{\mathrm{eff}}}\right)^2 2(2N_{\mathrm{eff}})\sigma_x^4 = \frac{1}{N_{\mathrm{eff}}}\sigma^4. \tag{A-29}$$

Using the definition of N_{eff} allows the standard deviation of the ensemble autocorrelation to be expressed in terms of the time-bandwidth product,

$$\sigma_\alpha = \frac{\sigma^2}{\sqrt{N_{\mathrm{eff}}}} = \frac{\sigma^2}{\sqrt{BT+1}} = \frac{\sigma^2}{\sqrt{\frac{B(N-1)}{f_S}+1}}. \tag{A-30}$$

Finally, to illustrate the convergence of the χ_N^2 distribution to a Gaussian, consider the plot shown in Figure A-3 for the situation wherein 150 complex-valued samples are averaged. Note the close convergence even with this relatively moderate sample size.

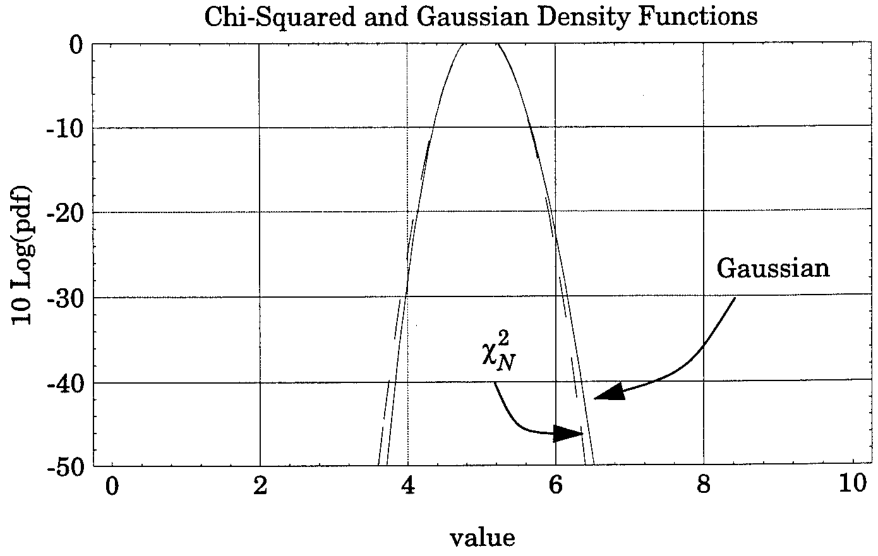


Figure A-3: Convergence of the χ_N^2 Distribution to the Gaussian ($M = 150$)

A.4 The Cross-Correlation Statistics

The off-diagonal terms in $\boldsymbol{ss}^\dagger$ and $\boldsymbol{nn}^\dagger$ as well as all of the terms of $\boldsymbol{sn}^\dagger$ and $\boldsymbol{ns}^\dagger$ will be the product of two complex-valued random variables—which, for the purposes of this analysis, are assumed to be independent band-limited Gaussian. In this section we examine the statistics associated with a single sampling epoch as well as the ensemble statistics generated by averaging over the sampling interval. Note that while the autocorrelation is real valued, these terms are, in general, complex-valued.

A.4.1 The Cross-Correlation Statistics at a Given Epoch

Let us denote z_i and z_j as the two complex-valued data channels. The expected value of their cross-product is,

$$E(z_i z_j^\dagger) = E((x_i + \iota y_i)(x_j - \iota y_j)) = E((x_i x_j + y_i y_j) + \iota(x_j y_i - x_i y_j)) = 0. \qquad \text{(A-31)}$$

More insight may be garnered by adopting a polar notations, thus,

$$E(z_i z_j^\dagger) = E(r_i r_j e^{\iota(\theta_i - \theta_j)}) = E(r_i r_j e^{\iota\Delta\theta}) = 0 \qquad \text{(A-32)}$$

where since θ_j is uniformly distributed from 0 to 2π, the difference $\theta_1 - \theta_2$ must also be uniformly distributed—leading to the expected value of zero. Note, however, that this is actually an ensemble average since, at any *given* sampling epoch, the sampled value *will not* be zero! In this case, we would have a phasor with an expected radius of

$$E(r_i r_j) = \int_0^\infty \int_0^\infty r_i r_j f_{r_i} f_{r_j} dr_i dr_j = E(r_i)E(r_j) = \frac{\pi\sigma_i\sigma_j}{4} \qquad \text{(A-33)}$$

which is simply the product of the expected values of r_1 and r_2 using the assumption of independent random variables. This phasor would be uniformly distributed from 0 to 2π. The variance of the product radius about this expected value can be shown to be

$$E((r_i r_j)^2) = \sigma_i^2\sigma_j^2 \qquad \text{(A-34)}$$

so that the standard deviation is,

$$\sigma_{r_i r_j} = \sigma_i\sigma_j\sqrt{1 - \frac{\pi^2}{16}}. \qquad \text{(A-35)}$$

As we add more samples in building the sampled covariance matrix, we will create a "drunkard's walk" in the complex-plane which—upon averaging—will converge to the expected value of Eqn (A-31). The characteristics of the ensemble average is the topic of the following section.

A.4.2 The Statistics of the Ensemble Average of the Cross-Correlation Terms

At any given sampling epoch, we will have a phasor characterized by a radius with an expected value of Eqn (A-33) and a standard deviation given in Eqn (A-35). This phasor is uniformly distributed over the range from 0 to 2π; thus, each epoch represents *two* independent samples. Expressing this in Cartesian (x, y) coordinates leads to representing the sample at the n^{th} epoch as,

$$x_n = R_n \cos(\Delta\theta_n) \qquad \text{(A-36)}$$

$$y_n = R_n \sin(\Delta\theta_n) \qquad \text{(A-37)}$$

where,

$$R_n = r_i r_j . \tag{A-38}$$

Assuming we average over the N complex-sample data set which contains N_{eff} effective samples, then the expected value of the ensemble average is zero.

$$E(\bar{x}) = E(\sum_{n=1}^{N} \frac{x_n}{N_{\text{eff}}}) = \frac{1}{N_{\text{eff}}} \sum_{n=1}^{N} E(R_n \cos\Delta\,\theta_n) = E(R)E(\cos\Delta\,\theta) = 0 \tag{A-39}$$

$$E(\bar{y}) = E(\sum_{n=1}^{N_{\text{eff}}} \frac{1}{N_{\text{eff}}} y_n) = \frac{1}{N_{\text{eff}}} \sum_{n=1}^{N_{\text{eff}}} E(R_n \sin\,(\Delta\,\theta_n)) = E(R)E(\sin\Delta\,\theta_n) = 0 \tag{A-40}$$

Note that just as the real and imaginary portions of the epoch values are independent, the ensemble average components are independent. Continuing to assume that the data set is composed of replicas of N_{eff} independent samples leads to the variance of the components being

$$\sigma_{\bar{x}}^2 = E(\bar{x}^2) = E(\sum_{n=1}^{N} \left(\frac{x_n}{N_{\text{eff}}}\right)^2) = \frac{1}{N_{\text{eff}}^2} \sum_{n=1}^{N} E((R_n \cos\Delta\,\theta_n)^2) = \frac{E(R^2)E(\cos^2\Delta\,\theta)}{N_{\text{eff}}} = \frac{\sigma_i^2\sigma_j^2}{2N_{\text{eff}}} \tag{A-41}$$

and

$$\sigma_{\bar{y}}^2 = E(\bar{y}^2) = E(\sum_{n=1}^{N} \left(\frac{y_n}{N_{\text{eff}}}\right)^2) = \frac{1}{N_{\text{eff}}^2} \sum_{n=1}^{N} E((R_n \sin\Delta\,\theta_n)^2) = \frac{E(R^2)E(\sin^2\Delta\,\theta)}{N_{\text{eff}}} = \frac{\sigma_i^2\sigma_j^2}{2N_{\text{eff}}} \tag{A-42}$$

where we should recall that σ_i^2 and σ_j^2 are the power spectral densities associated with the two band-limited Gaussian noise sources. In examining Eqn (A-41) and Eqn (A-42), note that the real and imaginary variances are equal—which is to be expected due to the initial assumption of independence. Finally, adopting a polar notation, and assuming enough samples so that the distributions may be modelled as Gaussian (i.e., the central limit theorem applies), the averaged cross-correlation may be modelled as a radial term, $\bar{R}$ having a Rayleigh distribution with a standard deviation of

$$\sigma_{\bar{R}} = \frac{\sigma_i\sigma_j}{\sqrt{2N_{\text{eff}}}} \tag{A-43}$$

with this radius uniformly distributed from 0 to 2π. The factor of $\sqrt{2}$ in Eqn (A-44) is due to the Rayleigh distribution being defined in terms of the standard deviation of the x and y Gaussians.

A.4.3 Cross-Correlation with a Deterministic Signal

If we are interested in the statistical behavior of the interaction of the noise with a deterministic waveform (e.g., a cisoid), the analysis of the previous section holds with the expected value being zero and the standard deviation being

$$\sigma_{ij}^2 = \frac{1}{N_{\text{eff}}}\sigma_i^2 E(s^\dagger s) \tag{A-44}$$

where $E(s^\dagger s)$ is the power of the deterministic waveform. We can view this as a projection of the N_{eff}-dimensional noise signal onto the 1-dimensional space of the waveform. Thus, equivalent to the operations of a discrete Fourier transform, $(1/N_{\text{eff}})^{\text{th}}$ of the energy of the random and uniformly distributed noise would be expected to be aligned with the reference waveform. Note that this is the same result achieved in Eqn (A-43)! Hence, we see that the deterministic signal may be any waveform and not restricted to any mathematically aesthetic form such as a cisoid. In general, if two processes (signals) are interacting, the number of independent random samples is determined by the random variable with the maximum bandwidth.

A.5 Covariance Matrix Modelling and Data Simulation

Using the results from the previous section, we can simulate sampled covariance matrices by melding the deterministic effects of the array geometry with the statistical characterization of the signal and noise interactions. Hence, the sampled covariance matrix may be modelled as

$$\hat{\boldsymbol{R}} = \boldsymbol{D}\hat{\boldsymbol{R}}_s\boldsymbol{D}^\dagger + \boldsymbol{D}\hat{\boldsymbol{R}}_{sn} + \hat{\boldsymbol{R}}_{sn}{}^\dagger\boldsymbol{D}^\dagger + \hat{\boldsymbol{R}}_n \tag{A-45}$$

where the elements of $\hat{\boldsymbol{R}}_s$, $\hat{\boldsymbol{R}}_{sn}$, and, $\hat{\boldsymbol{R}}_n$ are random variables characterized by the results of the previous sections and $\boldsymbol{D}$ is the steering matrix defined by the source–array geometry.

In conclusion, note that the derivations in this chapter presumed *narrowband* signals impinging on the array. If this assumption is seriously violated, the steering matrix and covariance matrix decomposition of Eqn (A-1) will be erroneous. Under such a scenario, a spectral decomposition of the model such that the narrowband assumption holds would be appropriate.

Appendix B. Subspace Stability Implementation

This chapter contains software written in the *Mathematica* programming language which illustrate the implementation of the subspace stability tracking concepts. In many respects, the subspace stability methods are unique due to their *algorithmic* rather than *mathematical* nature. This algorithmic nature yields the robust enumeration and DOA estimation performance displayed by the SSET against actual data. The function listings in this section are intended to complement the algorithmic descriptions of the root-Tracker described in Section 7.5.1 and the SSET described in Section 7.6.2.

B.1 Track Structure

During the course of data processing, the algorithms create and delete tracks; each track represents a potential signal source. The structure of a track contains the following fields:

avg	complex-valued estimated root location
stdDev	real-valued sampled standard deviation of the roots.
gateSize	circular gate size used for data association
state	track status: either “confirmed”, “probable”, or “tentative” depending upon confirming evidence
consHits	consecutive epochs in which data has been associated to the track
consMiss	consecutive epochs during which no roots have been associated to the track
startTime	initial epoch resulting in a track
duration	span during which the track has existed
obsList	list of the roots which are used in the calculation of the *avg* and *stdDev*

This track structure implicitly assumes a stationary geometry—i.e., a constant spatial (or spectral) frequency.

B.2 Nearest-Neighbor Data Association

The nearest-neighbor data association algorithm assumes a monotonic mapping so that each root can be associated with one-and-only-one track and each track can be associated with only one root from a given epoch. The association is accomplished by gating the roots and assigning unique associations sequentially from closest association to furthest.

```
NearestNeighborMap[roots_List, trackSet_List] := Module[
   {rootMap, assocMap, B, obs, trk, distance,
   avg=1, stdDev=2, gateSize=3},

   rootMap = {};
   Do[ (* for each root in the set, check the existing track association *)
      Do[ (* check the root distance from existing tracks for this root *)
         distance = Abs[roots[[obs]] - trackSet[[trk,avg]]];
         If[ distance <= trackSet[[trk,gateSize]],
            AppendTo[rootMap, {obs, trk, distance}]];
      ,{trk,1,Length[trackSet]}];
   ,{obs,1,Length[roots]}];

   rootMap = Sort[rootMap,(#1[[3]] < #2[[3]]&)]; (* sort by distance *)
   assocMap = {};  (* select unique maps having shortest distance *)
   While[Length[rootMap] > 0,
      AppendTo[assocMap, B = rootMap[[1]]];
      rootMap = Select[rootMap,(#[[1]] != B[[1]]) && (#[[2]] != B[[2]])&];
   ];
   assocMap
   ];
```

B.3 Root-Tracker Function

The root-Tracker algorithm is conceptual sister of Burkhardt's method. Thus, for a K-element array, each sampling interval will result in $K-1$ complex-valued roots and the root-Tracker sequentially processes the roots looking for root stability. Under the classical assumptions, the signal roots will be stable and reside on the unit-circle whereas the spurious noise-induced roots will be temporally uncorrelated. As noted in Section 7.5, these assumptions are frequently violated in practice; as a result, the SSET algorithms should generally be preferred.

This implementation operational parameters (Options[rootTracker) are:

initialGate	initial *gateSize* to be used for root-to-track association
minimumGate	minimum allowable *gateSize*
num2promote	number of sequential associations required for track promotion to “confirmed” status
num2delete	number of sequential association misses required for track deletion
filterSpan	the number of observations which should be maintained and used in the parameter estimation
gateTimeConstant	the first-order filter time constant used in adjusting the *gateSize* to prevent premature gate closure

This implementation implicitly assumes a stationary geometry. Geometry dynamics are primarily reflected in the choice of the *filterSpan* which defines the amount of data used to establish the estimated DOA.

```
rootTracker[{epoch_, newRoots_List}, initTrack_List, opts___Rule] := Module[
   {numRoots = Length[newRoots], track, trk, obs, distance, maxPoints,
   decay, b0, a1, initGate, minGate, B,
   rootMap, assocMap, deletedTracks, promoteThreshold, deleteThreshold,
   associatedRoots, associatedTracks, freeRoots, freeTracks,
   avg=1, stdDev=2, gateSize=3, state=4, (* track indices *)
   consHits=5, consMiss=6, startTime=7, duration=8, obsList=9},

   maxPoints = filterSpan /. {opts} /. Options[rootTracker];
   decay = gateTimeConstant /. {opts} /. Options[rootTracker];
   minGate = minimumGate /. {opts} /. Options[rootTracker];
   initGate = initialGate /. {opts} /. Options[rootTracker];
   promoteThreshold = num2promote /. {opts} /. Options[rootTracker];
   deleteThreshold = num2delete /. {opts} /. Options[rootTracker];

   a1 = Exp[-1.0/decay];
   b0 = 1.0 - a1;

   track = initTrack;
   assocMap = NearestNeighborMap[newRoots,track];

      (* the roots and tracks and their associations or lack thereof *)
```

```
associatedRoots  = Sort[Map[#[[1]]&, assocMap]];
associatedTracks = Sort[Map[#[[2]]&, assocMap]];
freeRoots  = Complement[Range[numRoots],associatedRoots];
freeTracks = Complement[Range[Length[track]],associatedTracks];

   (* update the tracks *)
Do[  (* work through the association map and update the tracks *)
   obs = assocMap[[m,1]];
   trk = assocMap[[m,2]];

   If[Length[track[[trk,obsList]]] >= maxPoints,
     track[[trk,obsList]] = Rest[track[[trk,obsList]]]];
   AppendTo[track[[trk,obsList]], newRoots[[obs]]];
   track[[trk,avg]] = Mean[track[[trk,obsList]]];
   track[[trk,stdDev]] = Sqrt[ComplexVariance[track[[trk,obsList]]]];
   track[[trk,gateSize]] = Max[ minGate,
           b0*(3 track[[trk,stdDev]]) + a1*track[[trk,gateSize]]];

   track[[trk, consHits]] += 1;
   track[[trk, consMiss]] = 0;
   track[[trk, duration]] = epoch - track[[trk, startTime]];
   If[ (track[[trk, state]] == "tentative") &&
        (track[[trk, consHits]] >= promoteThreshold),
     track[[trk, state]] = "confirmed";
     ];
,{m,1,Length[assocMap]}];

deletedTracks = {};
Do[  (* update and delete, if necessary, undetected tracks *)
   trk = freeTracks[[m]];
   track[[trk, consHits]] = 0;
   track[[trk, consMiss]] += 1;
   Switch[track[[trk, state]],
     "confirmed",
        If[track[[trk, consMiss]] >= deleteThreshold,
           track[[trk, state]] = "deleted";
           AppendTo[deletedTracks, track[[trk]]];
           track = Delete[track,trk];
           ],
     "tentative",
        track = Delete[track,trk];
     ];
,{m,Length[freeTracks],1,-1}];  (* work from end to beginning *)

Do[  (* update the track list with new tentative tracks *)
   obs = freeRoots[[m]];
```

```
  newTrack = Range[obsList]; (* initialize list *)

      (* initialize the new addition to the track set *)
  newTrack[[avg]] = newRoots[[obs]];
  newTrack[[stdDev]] = 0;
  newTrack[[gateSize]] = initialGate /.{opts} /. Options[rootTracker];
  newTrack[[state]] = "tentative";
  newTrack[[consHits]] = 1;
  newTrack[[consMiss]] = 0;
  newTrack[[startTime]] = epoch;
  newTrack[[duration]] = 0;
  newTrack[[obsList]] = {newRoots[[obs]]};

  AppendTo[track,newTrack]; (* append to the track list *)
,{m,1,Length[freeRoots]}];

track  (* return the confirmed and tentative tracks *)
]
```

B.4 SSET Function

This implementation of the SSET (Subspace Stability Exploitation Tracker) algorithm uses the *MultHypGEESEroots* function to generate sets of roots from a given covariance matrix; each root set corresponds to a hypothesized number of signals, $\hat{J}$, under the constraint that hypotheses must result in *all* roots residing near the unit-circle. If existing tracks are available, the new roots are processed with the hypothesis selected whose roots best correspond to existing tracks. If no tracks are available, the root history is compiled and processed via the *SSETrackerInit* function until a stable signal source is identified.

This implementation operational parameters (Options[SSET]) are:

initialGate	initial *gateSize* to be used for root-to-track association
minimumGate	minimum allowable *gateSize*
num2promote	number of sequential associations required for track promotion to "confirmed" status
num2delete	number of sequential association misses required for track deletion
filterSpan	the number of observations which should be maintained and used in the parameter estimation

gateTimeConstant	the first-order filter time constant used in adjusting the *gateSize* to prevent premature gate closure
validRadiusOffset	valid hypothesized roots must lie within ±*validRadiusOffset* of the unit-circle

As with the root-Tracker, a static source-array geometry is implicitly assumed. If the hypothesis which best fits established tracks results in unassociated roots, the new tracks derived from these roots are designated "probable" since there is a reasonable likelihood that they represent short-term events.

```
SSET[epoch_, vecs_List, currentTracks_, oldRoots_, opts___Rule] := Module[
   {tracks, rootSets, hypSet, hypIndex, promoteThreshold, trk, obs, m,
   costSet, roots, assocMap, cumDistance, numMatches, maxMatch, bestMatch,
   newTrack,
   avg=1, stdDev=2, gateSize=3, state=4, (* track indices *)
   consHits=5, consMiss=6, startTime=7, duration=8, obsList=9},

           (* track processing parameters *)
   maxPoints = filterSpan /. {opts} /. Options[SSET];
   decay = gateTimeConstant /. {opts} /. Options[SSET];
   minGate = minimumGate /. {opts} /. Options[SSET];
   initGate = initialGate /. {opts} /. Options[SSET];
   promoteThreshold = num2promote /. {opts} /. Options[SSET];
   deleteThreshold = num2delete /. {opts} /. Options[SSET];

   a1 = Exp[-1.0/decay];
   b0 = 1.0 - a1;

   tracks = currentTracks;(* existing tracks -- all categories *)
   rootSets = oldRoots;(* root history -- used for track init *)

   hypSet = MultHypGEESEroots[vecs]; (* possible root sets *)

   If[Length[tracks] == 0, (* no tracks => attempt initialization *)
      (* note that the empty hypothesis set is implicitly handled *)
      AppendTo[rootSets, hypSet]; (* update the history *)
      If[Length[rootSets] >= promoteThreshold,
         rootSets = Take[rootSets,-promoteThreshold];
         tracks = SSETrackerInit[rootSets];
         If[Length[tracks] > 0,
            Do[  (* set track start times to current epoch *)
               tracks[[trk, startTime]] = epoch;
         ,{trk,1,Length[tracks]}];
         rootSets = {}; (* empty the history buffer *)
         ];
      ];
,(* we have existing confirmed tracks so check for best fit *)
   If[Length[hypSet] > 0, (* we have roots to analyze *)
      costSet = {};
      Do[ (* compare the roots to the existing tracks *)
         roots = hypSet[[hypIndex]];
         assocMap = NearestNeighborMap[roots, tracks];
         If[Length[assocMap] > 0,
            cumDistance = Plus@@Map[#[[3]]&,assocMap];
            numMatches = Length[assocMap];
            AppendTo[costSet,
                     {hypIndex,numMatches,cumDistance,assocMap}];
            ];
      ,{hypIndex,1,Length[hypSet]}];

      If[Length[costSet] > 0, (* we have viable associations *)
         maxMatch = Max[Map[#[[2]]&,costSet]]; (* max associations *)
         costSet = Select[costSet,#[[2]] == maxMatch&];

            (* given the identified best fit, update the track *)
         bestMatch = Sort[costSet, #1[[3]] < #2[[3]]&][[1]];
         roots = hypSet[[bestMatch[[1]]]];
         assocMap = bestMatch[[4]];

          (* roots & tracks and their associations or lack thereof *)
         associatedRoots  = Sort[Map[#[[1]]&, assocMap]];
         associatedTracks = Sort[Map[#[[2]]&, assocMap]];
         freeRoots  =
               Complement[Range[Length[roots]],associatedRoots];
         freeTracks =
               Complement[Range[Length[tracks]],associatedTracks];

      ,(* we have viable roots but no associations *)
         assocMap = {};
         associatedRoots  = {};
         associatedTracks = {};
         roots = Last[hypSet]; (* select maximum hypothesis *)
         freeRoots  = Range[Length[roots]];
         freeTracks = Range[Length[tracks]];
      ];

    ,(* the hypSet is empty so we don't have roots to process *)
      assocMap = {};
   associatedRoots  = {};
   associatedTracks = {};
   freeRoots  = {};
   freeTracks = Range[Length[tracks]];
]; (* end hypothesis processing *)

(* having considered the possible associations, update the tracks *)

Do[  (* work through the association map and update the tracks *)
   obs = assocMap[[m,1]];
   trk = assocMap[[m,2]];

      If[Length[tracks[[trk,obsList]]] >= maxPoints,
      tracks[[trk,obsList]] = Rest[tracks[[trk,obsList]]]];
   AppendTo[tracks[[trk,obsList]], roots[[obs]]];
   tracks[[trk,avg]] = Mean[tracks[[trk,obsList]]];
   tracks[[trk,stdDev]] =
                Sqrt[ComplexVariance[tracks[[trk,obsList]]]];
   tracks[[trk,gateSize]] = Max[ minGate,
         b0*(3 tracks[[trk,stdDev]]) + a1*tracks[[trk,gateSize]]];
   tracks[[trk, consHits]] += 1;
   tracks[[trk, consMiss]] = 0;
   tracks[[trk, duration]] = epoch - tracks[[trk, startTime]];
   If[ (tracks[[trk, state]] == "probable") &&
         (tracks[[trk, consHits]] >= promoteThreshold),
      tracks[[trk, state]] = "confirmed";
      ];
,{m,1,Length[assocMap]}];

Do[  (* update and delete, if necessary, undetected tracks *)
   trk = freeTracks[[m]];
   tracks[[trk, consHits]] = 0;
   tracks[[trk, consMiss]] += 1;
   Switch[tracks[[trk, state]],
      "confirmed",
         If[tracks[[trk, consMiss]] >= deleteThreshold,
            tracks = Delete[tracks,trk];
            ],
      "probable",
         tracks = Delete[tracks,trk],
      "tentative",
         tracks = Delete[tracks,trk];
      ];
,{m,Length[freeTracks],1,-1}];  (* work from end to beginning *)

Do[  (* update the track list with new probable tracks *)
```

```
        obs = freeRoots[[m]];
        newTrack = Range[obsList]; (* initialize list *)

              (* initialize the new addition to the track set *)
        newTrack[[avg]] = roots[[obs]];
        newTrack[[stdDev]] = 0;
        newTrack[[gateSize]] = initialGate /.{opts} /. Options[SSET];
        newTrack[[state]] = "probable";
        newTrack[[consHits]] = 1;
        newTrack[[consMiss]] = 0;
        newTrack[[startTime]] = epoch;
        newTrack[[duration]] = 0;
        newTrack[[obsList]] = {roots[[obs]]};

        AppendTo[tracks,newTrack]; (* append to the track list *)
      ,{m,1,Length[freeRoots]}];

      ];
   {tracks, rootSets}
   ];
```

B.5 MultHypGEESEroots Function

Given a set of eigenvectors, *vecs*, derived from a sampled array covariance matrix, this function applies the GEESE DOA estimation algorithm under a sequence of hypothesized number of signals. The sets of roots associated with valid hypotheses are returned. A valid hypothesis is defined as one in which *all* resulting roots are within ±*validRadiusOffset* of the unit-circle.

```
MultHypGEESEroots[vecs_?MatrixQ, opts___Rule] := Module[
   {rootSet, validHypothesis, hypRoots, j, threshold},

   threshold = validRadiusOffset /. {opts} /. Options[SSET];

   rootSet = {};(* sets associated with valid roots *)
   validHypothesis = True;(* valid hypothesis flag *)
   j = 1;          (* current hypothesis *)

   While[ validHypothesis && j < Length[vecs],(* i.e., reasonable "j" *)
      hypRoots = GEESE[vecs, j, rootForm->"complex"];
      If[Max[Abs[Abs[#]-1]& /@ hypRoots] <= threshold,
         rootSet = Append[rootSet,hypRoots];
         j += 1;
      , (* otherwise, stop processing *)
```

```
         validHypothesis = False
         ]
      ];
   rootSet
   ];
```

B.6 SSETrackerInit

This function is used by the *SSET* algorithm for track initialization if no existing tracks are available for use as a reference subspace. Given a sequence of potential root sets derived by the *MultHypGEESEroots* function, the common enumeration hypotheses for each data segment are identified The associated data sets are then analyzed for track consistency under a *common* hypothesis—that is, the same hypothesized number of signals for each segment. If the number of confirmed tracks equals the number of hypothesized signals, "confirmed" tracks are returned. In the event that multiple hypotheses satisfy the hypothesis confirmation, the largest hypothesis is used to generate the returned tracks.

```
SSETrackerInit[rootSets_List, opts___Rule] := Module[
   {promoteThreshold, epochs, minHyp, maxHyp, hyp, hypIndex,
   balancedSets, n, roots, ,trk, obs, m, preferredTrackSet,
   assocMap, associatedTracks, freeTracks, tracks, epochHyps,
   candidateHyps, trackSet,
   decay, minGate, initGate, numTracksMatchHypothesis,
   avg=1, stdDev=2, gateSize=3, state=4, (* track indices *)
   consHits=5, consMiss=6, startTime=7, duration=8, obsList=9},

   promoteThreshold = num2promote /. {opts} /. Options[SSET];
   decay = gateTimeConstant /. {opts} /. Options[SSET];
   minGate = minimumGate /. {opts} /. Options[SSET];
   initGate = initialGate /. {opts} /. Options[SSET];

   a1 = Exp[-1.0/decay];
   b0 = 1.0 - a1;

   epochs = Length[rootSets];

   epochHyps = {};
   Do[
      If[Length[rootSets[[n]]] > 0,
         AppendTo[epochHyps,Map[Length[#]&,rootSets[[n]]]];
      ,(* else we have a null set *)
         AppendTo[epochHyps,{0}];
      ];
    ,{n,1,Length[rootSets]}];
    candidateHyps = Intersection[Sequence@@epochHyps];

    If[Length[candidateHyps] > 0 && Min[candidateHyps] > 0,
                            (* we have hypotheses to check *)
      minHyp = Min[candidateHyps]; (* minimum "J" *)
      maxHyp = Max[candidateHyps]; (* maximum "J" *)
      balancedSets =
        Map[Select[#,minHyp <= Length[#] <= maxHyp&]&,rootSets];

      trackSet = Range[Sequence@@candidateHyps];
      Do[(* iteratively process the common hypotheses *)

           (* initialize the tracker *)
        roots = balancedSets[[1,hypIndex]];
        tracks = trackInit[roots];

        Do[(* process the epochs for consistency *)
          roots = balancedSets[[n,hypIndex]];
          assocMap = NearestNeighborMap[roots, tracks];

            (* associated and unmapped tracks *)
          associatedTracks = Sort[Map[#[[2]]&, assocMap]];
          freeTracks =
              Complement[Range[Length[tracks]],associatedTracks];

          Do[  (* work through association map and update tracks *)
            obs = assocMap[[m,1]];
            trk = assocMap[[m,2]];

            AppendTo[tracks[[trk,obsList]], roots[[obs]]];
            tracks[[trk,avg]] = Mean[tracks[[trk,obsList]]];
            tracks[[trk,stdDev]] =
                    Sqrt[ComplexVariance[tracks[[trk,obsList]]]];
            tracks[[trk,gateSize]] = Max[ minGate,
              b0*(3 tracks[[trk,stdDev]]) +
                a1*tracks[[trk,gateSize]]];
            tracks[[trk, consHits]] += 1;
            tracks[[trk, consMiss]] = 0;
            tracks[[trk, duration]] += 1;
            If[ (tracks[[trk, state]] == "tentative") &&
                  (tracks[[trk, consHits]] >= promoteThreshold),
              tracks[[trk, state]] = "confirmed";
              ];
          ,{m,1,Length[assocMap]}];
```

```
            Do[  (* delete unassociated tracks *)
               trk = freeTracks[[m]];
               tracks = Delete[tracks,trk];
            ,{m,Length[freeTracks],1,-1}];  (* work end to beginning *)

         ,{n,2,epochs}];
         trackSet[[hypIndex]] = tracks; (* persistent tracks *)
      ,{hypIndex, Length[trackSet]}];

      numTracksMatchHypothesis = False;
      preferredTrackSet = {};
      hyp = maxHyp;
      While[ (!numTracksMatchHypothesis) && (hyp >= minHyp),
         hypIndex = hyp - minHyp + 1;
         If[Length[trackSet[[hypIndex]]] == hyp,
            preferredTrackSet = trackSet[[hypIndex]];
            numTracksMatchHypothesis = True;
         ,(* else *)
            hyp -= 1;
            ];
         ];
      Select[preferredTrackSet,#[[state]] == "confirmed"&]
   ,(* otherwise,  we don't have a confirmed track *)
      {}
   ]
   ]
```

B.7 trackInit Function

The *trackInit* utility function is used by the *SSET* and *SSETrackerInit* functions to initialize tracks based upon validated roots.

```
trackInit[rootSet_List] := Module[
   {rootNum, tracks, newTrack,
   avg=1, stdDev=2, gateSize=3, state=4, (* track indices *)
   consHits=5, consMiss=6, startTime=7, duration=8, obsList=9},

   tracks = {};
   Do[ (* for each root in the rootSet *)
      newTrack = Range[obsList]; (* initialize list *)

            (* initialize the new addition to the track set *)
      newTrack[[avg]] = rootSet[[rootNum]];
```

```
  newTrack[[stdDev]] = 0;
  newTrack[[gateSize]] = initialGate /. Options[SSET];
  newTrack[[state]] = "tentative";
  newTrack[[consHits]] = 1;
  newTrack[[consMiss]] = 0;
  newTrack[[startTime]] = 0;
  newTrack[[duration]] = 0;
  newTrack[[obsList]] = {rootSet[[rootNum]]};

  AppendTo[tracks,newTrack]; (* append to the track list *)
,{rootNum, 1, Length[rootSet]}];
tracks
];
```

References

Subspace Methods & Algorithms

[1] R. O. Schmidt, *A Signal Subspace Approach to Multiple Emitter Location and Spectral Estimation*, Ph.D. dissertation, Stanford University, 1981.

[2] S. U. Pillai, *Array Signal Processing*, Springer-Verlag, New York, 1989.

[3] B. H. Kwon, *New High Resolution Techniques and their Performance Analysis for Angles-of-Arrival Estimation*, Ph.D. dissertation, Polytechnic University, 1989.

[4] S. S. Reddi, "Multiple Source Location—A Digital Approach," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 15, no. 1, January 1979.

[5] R. Kumaresan and D. W. Tufts, "Estimating the Angles of Arrival of Multiple Plane Waves," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 19, no. 1, January 1983.

[6] R. Kumaresan, "On the Zeros of the Linear Prediction-Error Filter for Deterministic Signals," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 31, no. 1, February 1983.

[7] A. J. Barabell, "Improving the resolution performance of eigenstructure-based direction-finding algorithm," *Proceedings of the IEEE ICASSP'83*, pp. 336–339, 1983.

[8] R. Roy, A. Paulraj, and T. Kailath, "ESPRIT—A Subspace Rotation Approach to Estimation of Parameters of Cisoids in Noise," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. ASSP-34, no. 5, October 1986.

[9] R. H. Roy and T. Kailath, "ESPRIT—Estimation of Signal Parameters via Rotational Invariance Techniques", *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 37, no. 7, July 1989.

[10] M. D. Zoltowski and D. Stavrinides, "Sensor Array Signal Processing Via a Procrustes Rotations Based Eigenanalysis of the ESPRIT Data Pencil", *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 37, no. 6, June 1989.

[11] M. D. Zoltowski, "Solving the Generalized Eigenvalue Problem with Singular Forms", *Proceedings of the IEEE*, vol. 75, no. 11, November 1987.

[12] J. A. Cadzow, "Multiple Source Location—The Signal Subspace Approach", *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 38, no. 7, July 1991.

[13] M. Viberg and B. Ottersten, "Sensor Array Processing Based on Subspace Fitting", *IEEE Transactions on Signal Processing*, vol. 39, no. 5, May 1991.

Signal Detection via Eigenvalue Analysis

[14] M. Wax, *Detection and Estimation of Superimposed Signals*, Ph.D. dissertation, Stanford University, March 1985.

[15] M. Wax and T. Kailath, "Detection of Signals by Information Theoretic Criteria", *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. ASSP-33, no. 2, April 1985.

[16] L. C. Zhao, P. R. Krishaiah, and Z. D. Bai, "On Detection of the Number of Signals in Presence of White Noise", *Journal of Multivariate Analysis*, vol. 20, no. 1, October 1986.

[17] H. Wang and M. Kaveh, "On the Performance of Signal-Subspace Processing—Part I: Narrow-Band Systems", *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. ASSP-34, no. 5, October 1986.

[18] C. G. Khatri and C. R. Rao, "Effects of Estimated Noise Covariance in Optimal Signal Detection", *IEEE Transactions on Acoustic, Speech, and Signal Processing*, vol. ASSP-35, no. 5, May 1987.

[19] Y. Q. Yin and P. R. Krishnaiah, "On Some Nonparametric Methods for Detection of the Number of Signals", *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. ASSP-35, no. 11, November 1987.

[20] M. Kaveh, H. Wang, and H. Hung, "On the Theoretical Performance of a Class of Estimators of the Number of Narrow-Band Sources", *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. ASSP-35, no. 9, September 1987.

[21] M. Wax and I. Ziskind, "Detection of the Number of Coherent Signals by the MDL Principle", *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 37, no. 8, August 1989.

[22] Q. T. Zhang, K. M. Wong, P. C. Yip, and J. P. Reilly, "Statistical Analysis of the Performance of Information Theoretic Criteria in the Detection of the Number of Signals in Array Processing", *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 37, no. 10, October 1989.

[23] K. M. Wong, Q. T. Zhang, J. P. Reilly, and P. C. Yip, "On Information Theoretic Criteria for Determining the Number of Signals in High Resolution Array Processing", *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 38, no. 11, November 1990.

[24] W. Chen, K. M. Wong, and J. P. Reilly, "Detection of the Number of Signals: A Predicted Eigen-Threshold Approach", *IEEE Transactions on Signal Processing*, vol. 39, no. 5, May 1991.

[25] Q. Wu, and D. R. Fuhrmann, “A Parametric Method for Determining the Number of Signals in Narrow-Band Direction Finding”, *IEEE Transactions on Signal Processing*, vol. 39, no. 8, August 1991.

[26] J. W. Silverstein and P. L. Combettes, “Signal Detection via Spectral Theory of Large Dimensional Random Matrices”, *IEEE Transactions on Signal Processing*, vol. 40, no. 8, August 1992.

[27] J. Choi, I. Song, S. Kim, and S. Y. Kim, “A combined determination-estimation method for direction of arrival estimation”, *Signal Processing*, Elsevier, vol. 30, pp. 123–131, 1993.

[28] V. U. Reddy, and L. S. Biradar, “SVD-Based Information Theoretic Criteria for Detection of the Number of Damped/Undamped Sinusoids and Their Performance Analysis”, *IEEE Transactions on Signal Processing*, vol. 41, no. 9, September 1993.

Spatial Smoothing Algorithms and Analysis

[29] J. E. Evans, J. R. Johnson, and D. F. Sun, “High resolution angular spectrum estimation techniques for terrain scattering analysis and angle of arrival estimation,” *Proc. 1st ASSP Workshop on Spectral Estimation*, Hamilton, Ont., Canada, pp. 134–139, 1981.

[30] T. J. Shan, M. Wax, and T. Kailath, “On Spatial Smoothing for Direction-of-Arrival Estimation of Coherent Signals,” *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. ASSP-33, no. 4, August 1985.

[31] J. A. Cadzow, Y. S. Kim, and D. C. Shiue, “Resolution of coherent signals using a linear array”, *Proceedings of the ICASSP ’88*, pp. 1599–1600, IEEE, 1988.

[32] S. U. Pillai and B. H. Kwon, “Forward/Backward Spatial Smoothing Techniques for Coherent Signal Identification,” *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 37, no. 1, January 1989.

[33] Weixiu Du and R. L. Kirlin, “Improved Spatial Smoothing Techniques for DOA Estimation of Coherent Signals,” *IEEE Transactions on Signal Processing*, vol. 39, no. 5, May 1991.

[34] J. F. Yang and C. J. Tsai, “A Further Analysis of Decorrelation Performance of Spatial Smoothing Technique for Real Multipath Sources,” *IEEE Transactions on Signal Processing*, vol. 40, no. 8, August 1992.

[35] H. Krim and J. G. Proakis, “Smoothed Eigenspace-Based Parameter Estimation,”, *Automatica*, vol. 30, no. 1, pp. 27–38, January 1994.

Classical Methods of DOA Estimation

[36] L. A. Rubano, “Unambiguous Phase Measurement,” *ARL/PSU Patent Disclosure 575*, 4 February 1987

[37] Jack Capon, "High Resolution Frequency Wave Number Spectrum Analysis," *Proceedings of the IEEE*, vol. 57, no. 8, August 1969.

[38] J. P. Burg, "The relationship between maximum entropy spectra and maximum likelihood spectra," *Geophysics*, vol. 37, pp. 375–376, April 1972.

[39] V. F. Pisarenko, "The Retrieval of Harmonics from a Covariance Functions", *Geophys. J. Royal Astronomical Soc.*, No. 33, pp. 511–531, 1973.

[40] A. J. Berni, "Target Identification by Natural Resonance Estimation", *IEEE Transactions on Aerospace and Electronic Systems*, Vol. 11, pp. 147–154, 1975.

[41] G. V. Boriotti and L. J. Kaplan, "Superresolution of Uncorrelated Interference Sources by Using Adaptive Array Techniques," *IEEE Transactions on Antennas and Propagation*, vol. 27, no. 6, November 1979.

Array Calibration and Modelling Errors

[42] A. Paulraj and T. Kailath, "Direction of Arrival Estimation by Eigenstructure Methods with Unknown Sensor Gain and Phase", *Proceedings ICASSP '85*, 1985, pp. 640–643.

[43] A. Paulraj, T. J. Shan, V. U. Reddy, and T. Kailath, "A Subspace Approach to Determining Sensor Gain and Phase with Applications to Array Processing," *SPIE vol. 696 Advanced Algorithms and Architectures for Signal Processing*, 1986.

[44] A. J. Weiss and B. Friedlander, "Direction Finding in the Presence of Mutual Coupling", *Proceedings of the 1988 Asilomer Conference on Circuits, Systems, and Computers, pp. 598–602.*

[45] A. J. Weiss, A. S. Willsky, and B. C. Levy, "Eigenstructure Approach for Array Processing with Unknown Intensity Coefficients," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 36, no. 10, October 1988.

[46] A. J. Weiss and B. Friedlander, "Array Shape Calibration Using Sources in Unknown Locations—A Maximum Likelihood Approach," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 37, no. 12, December 1989.

[47] Daniel R. Fuhrmann, "Estimation of Sensor Gain and Phase", *IEEE Transactions on Signal Processing*, vol. 42, no. 1, January 1994.

[48] E. K. L. Hung, "A Critical Study of a Self-Calibrating Direction Finding Method for Arrays," *IEEE Transactions on Signal Processing*, vol. 42, no. 2, February 1994.

[49] V. C. Soon, L. Tong, Y. F. Huang, and R. Liu, "A Subspace Method for Estimating Sensor Gains and Phases," *IEEE Transactions on Signal Processing*, vol. 42, no. 4, April 1994.

Processing in Correlated Noise Fields

[50] A. Paulraj and T. Kailath, "Eigenstructure Methods for Direction of Arrival Estimation in the Presence of Unknown Noise Fields", *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol ASSP-34, no. 1, February 1986.

[51] J. P. Le Cadre, "Parametric Methods for Spatial Signal Processing in the Presence of Unknown Colored Noise Fields", *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 37, no. 7, July 1989.

[52] M. Wax, "Detection and Localization of Multiple Sources in Noise with Unknown Covariance", *IEEE Transactions on Signal Processing*, vol. 40, no. 1, January 1992.

[53] J. J. Fuchs, "Estimation of the Number of Signals in the Presence of Unknown Correlated Sensor Noise", *IEEE Transactions on Signal Processing*, vol. 40, no. 5, May 1992.

[54] Q. T. Zhang and K. M. Wong, "Information Theoretic Criteria for the Determination of the Number of Signals in Spatially Correlated Noise", *IEEE Transactions on Signal Processing*, vol. 41, no. 4, April 1993.

Algorithm Performance Analyses

[55] Fu Li, *A Unified Performance Analysis of Subspace-based DOA Estimation Algorithms in Signal Processing*, Ph.D. dissertation, University of Rhode Island, 1990.

[56] G. Bienvenu and L. Kopp, "Adaptivity to Background Noise Spatial Coherence for High Resolution Passive Methods", *Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Processing* (Denver, CO), pp. 307–310, 1980.

Dynamical Systems Approach

[57] Steven T. Smith, "Dynamical systems that perform the singular value decomposition", *Systems & Control Letters*, vol. 16, pp. 319–327, 1991.

[58] R. W. Brockett, "Dynamical Systems that Learn Subspaces", *Mathematical Systems Theory: The Influence of R. E. Kalman*, Springer-Verlag, Berlin, 1990.

Subspace Stability Exploitation

[59] R. C. Burkhardt, *A High Resolution Bearing Estimation Algorithm based upon the Concepts of MUSIC*, ARL/PSU TM 20-215, 31 July 1990.

[60] G. W. Stewart, "Stochastic Perturbation Theory", *SIAM Review*, vol. 32, no. 4, December 1990.

[61] S. D. Silverstein, "A New Use of ESPRIT for Detection and Model Order Estimation," *IEEE Signal Processing Letters*, vol. 1, no. 10, October 1994.

Multiple Target Tracking

[62] Samuel S. Blackman, *Multiple-Target Tracking with Radar Application*, Artech House, Norwood, MA, 1986.

[63] Yaakov Bar-Shalom and Thomas E. Fortmann, *Tracking and Data Association*, Academic Press, San Diego, CA, 1988.

General References

[64] R. A. Horn and C. R. Johnson, *Matrix Analysis*, Cambridge University Press, 1990.

[65] William S. Burdic, *Underwater Acoustic System Analysis*, Prentice-Hall, Englewood Cliffs, NJ, 1984.

[66] Louis L. Scharf, *Statistical Signal Processing: Detection, Estimation, and Time Series Analysis*, Addison-Wesley Publishing Co., Reading Massachusetts, 1991.

[67] H. J. Larson and B. O. Shubert, *Probabilistic Models in Engineering Sciences, Volume I: Random Variables and Stochastic Processes*, John Wiley & Sons, 1979.

[68] Bernard D. Steinberg, *Principles of Aperture and Array System Design*, John Wiley & Sons, New York, NY, 1976.

[69] Robert A. Monzigo and Thomas W. Miller, *Introduction to Adaptive Arrays*, John Wiley & Sons, New York, NY, 1980.

[70] Robert W. Hornbeck, *Numerical Methods*, Quantum Publishers, New York, NY, 1975.